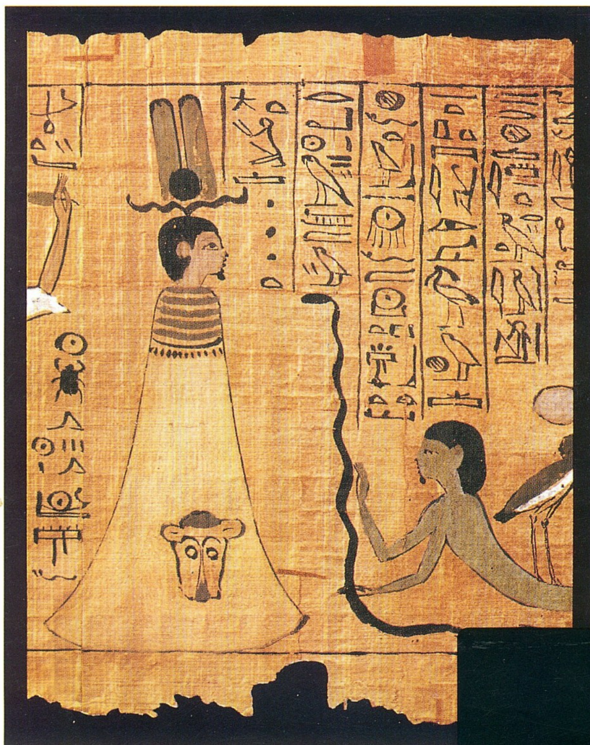
1993 / MÁJUS

ÁRA: 235 FT

ALAPLAP



MIKROSZÁMÍTÓGÉP MAGAZIN MÁGNESLEMEZ MELLÉKLETTEL



Elő a kártyákkal!

SWB — a májusi gyilkos

Mire jók a NOVRAM-ok?

Te lehetsz a szólista!

Ha lúd, legyen CD-ROM-on!

A HÓNAP TÉMÁJA:

A KÓDOLT ÜZENET

A MÁGNESLEMEZEN:

Mértékegység-átszámító
DES-titkosító — forráskódban
Táblázatgeneráló program
Egy lemezt egy fájlba
Játék Windows alatt

Munka a „pen-Windows" alatt

Keresetlen szavak a Disktoolról

Puhább közeg, keményebb feltételek

VIDEOTON
INFORMATIKA

IGÉNYLI
ÖN
A SZÉP FORMÁT?

A
MINŐSÉGET?



A
MEGBÍZHATÓSÁGOT?

AKKOR
VÁLASSZA

CHAMELEON VIDEOTERMINÁL

CSALÁDUNKAT

CHAMELEON–320 VIDEOTERMINÁL:

- DEC VT–320-kompatibilitás
- 14 inches, reflexiómentes képcső
- Dönthető, forgatható monitor
- Alacsony teljesítményfelvétel
- DEC VT–220-kompatibilis billentyűzet
- Rendelkezik a kötelező minősítésekkel



A CHAMELEON CSALÁD EGYÉB TAGJAI:

- Wyse-, Data General-, Kimtron KT 7 PC- stb. emulációkkal

8002 Székesfehérvár, Berényi út 100., Pf. 314 Telefon/Telefax: (22)319-013 Telefon: (22)312-730/25-02

INFORMÁCIÓKÉRÉS: A0524 ▼

ALAPLAP

Mikroszámítógép magazin
mágneslemez melléklettel
Alapította a Neumann János
Számítógéptudományi Társaság
és a Cédrus Informatikai Rt

Megjelenik havonta

Főszerkesztő:
Faklen Pál

Főszerkesztő-helyettes:
Varga János

Szerkesztők:
Jakab Ágnes
Sziebig Andrea

A Lemezkalauz és a Közkincs
rovat szerkesztője: Vékony Tamás

A szerkesztőbizottság tagjai:
Barna László, Broczkó Péter,
Brüll Károly, Csórián Sándor,
Farkas Ernő, Feleki Zoltán,
Fridl György, Herczeg József,
Kassay Árpád, Kónya László,
Kovács P. Attila, Lóth Tamás,
Sík Zoltán, Vargha Dénes,
Villányi László, Zoltai Péter

Szerkesztőség és kiadó:
1536 Budapest I., Márvány u. 17.
Telefonközpont: 156-3211
Fax: 156-9773
Hirdetésszervezési
telefon és fax: 175-0191

Kiadja az International Data Group
Magyarországi Lapkiadó Kft



Felelős kiadó: Bíró István
Műszaki vezető: Mészáros Tibor
Grafikai előkészítés:
IDG Grafikai Stúdió
Stúdióvezető: Lévai András
Szedés és formakészítés:
IDG Formakészítő Üzem
Vezető: Nemess József

Nyomtatás:
Zalai Nyomda, Zalaegerszeg
Felelős vezető: Galla József

Terjeszti a Magyar Posta,
az Extra-Hir és számos
számítástechnikai szaküzlet.
Előfizethető postautalvánnyal
a kiadónál (IDG Lapkiadó Kft,
1536 Budapest, Pf. 386), vagy
átutalással az IDG MKB 203-28016
pénzforgalmi jelzőszámra.

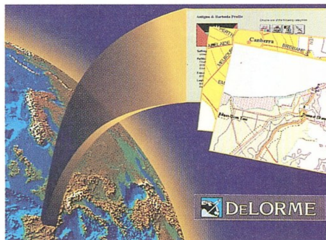
Példányonkénti eladási ár: 235 Ft
Évi előfizetési díj: 2 820 Ft

Külföldre terjeszti a Kultúra
H-1389 Budapest, Pf. 149

HU ISSN 0865-9788

A HÓNAP TÉMÁJA: A KÓDOLT ÜZENET

(Összeállította: Faklen Pál és Varga János)





HP LÉZERNYOMTATÓK AZ ALBACOMPTÓL



ALBACOMP SZÁMÍTÁSTECHNIKAI RÉSZVÉNYTÁRSASÁG

8000 Székesfehérvár, Hosszúsétatér 4-6.
Telefon: 22-315-414, Telefax: 22-327-532, Telex: 29 200 H

Color Plus

INFORMÁCIÓKÉRÉS: A0509 ▲

Kód és Dekód fia vagyok én...

Tévedés lenne azt hinni, hogy a kódolás csak titkosszolgálatok „ügye” — valójában minden, ami bennünket körülvesz, kódként viselkedik. Annyit mindenki tud, hogy a kódok alkalmazása a kódolás, annak visszaalakítása információvá pedig a dekódolás. A kód az az eszköz, amely valamilyen információ rögzítésére, továbbítására és visszaalakítására szolgál egyezményes jelrendszer segítségével.

Vannak nagy kultúrtörténeti kódrendszerek. Más mondanivalója van a Sixtusi Kápolna freskóinak például egy ateista, egy keresztény, egy zsidó, egy muzulmán vagy akár egy hindu számára. Még a temetők sírhantjai is megannyi információt hordoznak az elhunytakról. De csak azoknak mondják el titkaikat, akik neveltetésük okán — merthogy egyazon kultúrkörnyezetben élnek —, megtanulták dekódolni ezeket. Ha mi tévednénk be egy muzulmán sírkertbe, az számunkra valószínűleg kevésbé lenne beszédes, megfejthető.

Hogy mikor kódolunk? Mindig. Kód maga a beszéd is. Csak azok számára érthető, akik ismerik a dekódolásához szükséges kulcsot. A legtöbbünk által használt kódrendszer-rétegekből nehéz lenne összeválogatni az együvé tartozókat, de kétségtelenül van a kódrendszerekben valami „röghöz kötöttség”. Még egyetlen nyelv használatában is jól felismerhetők a helyi kódok. Amikor a beszédet leírjuk, az egyik kódrendszerből egy másikba tesszük át gondolatainkat. A kommunikációs eszközökben is ilyen folyamatok zajlanak le. A telefon a hangrezgéseket elektromos jelekké kódolja át, és ezeket egy másik eletronikus eszköz továbbkódolja. Az analóg beszédjelből valahol digitális jel lesz, majd a másik vonalvégen ismét beszéd.



Kódok özönében élünk. Amikor egy programmal vizsgálgatjuk a lemezt, nem is gondoljuk, hogy bűvészkedésünk a hexadecimális számokkal szintén egy átmeneti kódrendszer a mi jelrendszerünk és a gépek bináris rendszere között. Joggal kérdezhetjük az összeállítás egyik szerzőjével együtt: vajon a számítógépekhez miért éppen a bináris kódot választották, hiszen a hármas számrendszer hatékonyabb és közelebb is áll a természetes logaritmus alapszámához, az e-hez... Egyszer talán eszébe jut valakinek ilyen elven működő masinát is készíteni! Az alkatrészek ma már adottak, mert egy félvezető elemnek három állapota is lehet: nem vezet, vezet az egyik irányban és vezet a másik irányban. Elképzelhető, hogy ezt az elvet az emberi idegrendszer is alkalmazza.

A kódolás is lehet többféle. Az úgynevezett konvencionális kódrendszereknél a kódolás és a dekódolás folyamata közkincs, mindkét művelet elsajátítható a mindennapi kommunikációból, a nyilvános irodalomból, a dokumentumokból vagy például szabványok és ajánlások sokaságából. Van azonban olyan terület, ahol a kód nem hozzáférhető — ilyenkor beszélünk titkosításról.

A titkosítás, azaz a kriptológia is végigkísérte az emberiség kultúrtörténetét. Komoly tudós elmék munkáltak ki olyan nem szokványos kódrendszereket, amelyeket más tudós elmék szívós munkával igyekeztek megfejteni. Elvben — de csak elvben! — minden kód megfejthető. Csupán az a kérdés, hogy érdemes-e. Nem véletlen, hogy a kriptológiában egy kód hatékonyságát az abban a pillanatban létező leghatékonyabb megfejtési módszereknek a kód megfejtéséhez szükséges idejével mérik. Vannak esetek, amikor az emberiség birtokában lévő teljes számítástechnikai erőforrás összesített használatára lenne szükség ahhoz, hogy belátható időn belül hozzáférhetővé váljon a rejtjeles információ. És akkor a megfejtés már „nem ér annyit”.

A titkosítás tudományából a mindennapi üzleti élet is profitál. Titkosítva zajlik a banki kommunikáció, hasonló rendszerek védik a pénzátutalások biztonságát. Biztonsági kódok védik életünket, vagyonunkat. A kód, amely a fizikai fémkulcsot pótolja, megnyitja előttünk páncéltermek titkait. Ha elveszítettük vagy elfeledtük a titkos kódot, amely a széfet nyitja, rögtön az információtól elszigeteltek között találjuk magunkat. Hasonlóan titkos személyi kód, a PIN (personal indentification number) akadályozza meg, hogy hitelkártyánkról más vegyen fel pénzt a banki automatánál.

A zenészek, amikor a kottát olvassák, szintén dekódolnak. Kódok a mértékegységek is: megállapodáson alapuló, egyik kódrendszerből a másikba átvihető tartalommal bíró információt hordozó értékek. Kódok a számjegyeink, számrendszereink. Újabb és újabb szakmai és tudományos területek alakítják ki és bővítik a maguk sajátos jelrendszerét, így például a számítástechnika egészen rövid idő alatt terebélyes külön kis világot teremtett magának, amelyben a szakmán kívül álló idegenként mozog — mindaddig, amíg meg nem ismeri annak fogalmait, kódjait.

Kódrendszereink folyamatosan fejlődnek. A kódoló és visszafejtő műveletek sorát sajátítjuk el az iskolában is, amikor megtanulunk műszaki rajzot olvasni vagy a térképen eligazodni. Az életben pedig megtanuljuk, hogyan kell a dolgok rejtett értelmét kihámozni, a szavak mögött és a sorok között olvasni, a testbeszéd jeleit felfogni és értelmezni...

Kódolás és dekódolás az életünk.

Kis János

Mértékegységek átszámítása

Mi mennyi?

Mielőtt az olvasó belevetné magát a hónap témájának ez alkalommal keményebb agytornát követelő írásaiba, bemelegítésként ajánljuk az alábbi könnyű, lazító gyakorlatot.

A nyelvi eszközrendszerbe ágyazott kódok mellett egyik leggyakoribb „kódkörnyezetünket" a mértékegységek alkotják. Minden, ami mennyiségi ismérvekkel rendelkezik, szükségképpen mértékegységekkel is kódolt üzenet. Azok kódolása és dekódolása mindennapos feladat — mégsem mindig megy könnyen.

Tudományos szakszövegekben elmélyedve, de akkor is, ha csak egyszerű angol könyvet olvasunk, bajban vagyunk a mértékegységekkel. Hogyan számoljuk át azokat az általunk használtakra? Mert ha ezt nem tesszük meg, akkor tényleg nem tudjuk, hogy mi mennyi. Hiába vannak nemzetközi konvenciók, hiába az SI mértékegységrendszer, nehéz feladni a hagyományokat. (Mi is kilogrammban mérünk és nem kilopondban, vagy lóerőben gondolkodunk kilowatt helyett.)

Ezzel a gonddal mindenütt sokat küszködnek. Ezért készített Lars Josefsson svéd programozó is egy mértékegység-átszámító segédprogramot. Segítségével az általa ismert bármelyik mértékegységben megadott mennyiségi érték más mértékegységekre átszámított eredménye egy pillanat alatt leolvasható. A programot a tömörített állományból történő kicsomagolás után lehet elindítani. A bejelentkező képernyőn a klaviatúra nyíl gombjaival választhatjuk ki azt a szekciót — pl. tömeg, távolság stb. — ahová a kérdéses mértékegység tartozik. Utána már csak az a feladatunk, hogy a megfelelő mértékegységhez beírjuk a konvertálni kívánt értéket, és akkor az ablak összes többi mértékegységénél leolvashatjuk az átszámított értékeket. A visszalépés az ESC gombbal történik.

A legelső menüpont a help, amely elmondja, milyen formában írhatjuk be a kívánt mértékegységet. A normál mellett hatványkitevős formát is elfogad. Az F1 help mindig elérhető, de onnan érdemi információt nem lehet kapni, így használni sem érdemes. Ha azonban véletlenül betévedünk egy menübe, abból csak akkor tudunk kijönni, ha valamilyen adatot átszámoltattunk a programmal.

A program legalább 286-os processzort, MS-DOS 3.3-as vagy afölötti operációs rendszert igényel. Színes monitor nem szükséges, de ajánlott. Kilépéskor kiírja a program shareware jellegét ismertető szöveget, amit újabb ESC gombbal ugorhatunk át. A program lemezmellékletünkön önkicsomagoló tömörített formában a UNITS alkönyvtárban található.

Kis János

Kód a kódhoz

Időszámításunk előtt 300-ban Euklidész már tudta, hogy minden tetszőleges szám felbontható prímszámok szorzatára. A prímszámok — rendkívül egyszerű meghatározásuk ellenére — igen bonyolult problémákat hordoznak. Véletlenszerűen bukkannak fel, egyenként látszólag semmilyen törvényszerűségnek nem engedelmeskednek, eloszlásuk azonban meglepő rendszerességet mutat. Ők ugyanolyanok a „számkozmoszban", mint a kémiai elemek a természetben.

Az már régóta nyilvánvaló, hogy minél feljebb mászunk a számlétrán, annál ritkábban találkozhatunk prímekkel. 100 alatt még 25 prímszám van, tehát 25 százalék, az 1000 alatti számoknak már csak 16,8 százaléka prím, az 1 millió alattiaknak 7,8 százaléka, az 1 milliárd alattiaknak pedig csak 5,1 százaléka. A csökkenő tendencia tovább tart, de a prímszámok soha nem apadnak el teljesen, és ezt Euklidész elegáns gondolatmenettel bizonyította.

A prímeket azonban meg is kell találni, ami már nem olyan egyszerű. A keresésre 1645 tájékán egy párizsi pálos rendi szerzetes, Marin Mersenne talált olyan formulát, amit még napjainkban is gyakran használnak. Rájött, hogy ha kiszámítjuk kettőnek egy prímszámra (p) emelt hatványát és abból egyet levonunk, gyakran újra prímszámot kapunk. A 2^p-1 formula meglepő módon nemcsak a könnyen kiszámítható értékekre ($2^3-1 = 7$) bizonyult igaznak, hanem egészen nagy számokra is. Neki azonban még nem volt lehetősége tételét 20-30 jegyű számokra is ellenőrizni. 1876-ban Edouard Lucas bemutatta, hogy például a $2^{127}-1$ Mersenne-szám valóban egy 39 jegyű prímszám. Ez volt a legutolsó nem géppel kiszámolt legnagyobb prím, tehát megérdemli, hogy ide is leírjuk: 170 141 183 460 469 231 731 687 303 715 884 105 727. Csak 1951-ben sikerült ezt a Lucas-féle rekordot számítógéppel megdönteni.

Később már rendszerint Cray számítógépen dolgozó kutatók találták meg a mindenkori legnagyobb prímeket — ők is a Mersenne-képlettel dolgozva. 1985-ben David Slowinski egy Cray-X/MP gépen 3 óra alatt számította ki az akkori legnagyobb prímet ($2^{216\,091}-1$), amely 65 050 számjegyből áll, tehát ha az Alaplapban közölnénk, mintegy 10 oldalnyi terjedelmet foglalna el. A jelenleg általunk ismert legnagyobb prím 1992 márciusi keltezésű és szintén egy Mersenne-szám: $2^{756\,839}-1$.

Sokan azt gondolhatják, hogy a nagy prímszámokkal való foglalkozás öncélú játék. A következő oldalakon remélhetőleg minden érdeklődő számára kiderül, hogy ez nagyon is gyakorlatias kérdés. A „megfejthetetlen" kód ugyanis azon alapszik, hogy két nagy prímszám szorzata viszonylag egyszerűen létrehozható, de prímtényezőire igen nehezen, sőt a rendelkezésre álló idő alatt egyáltalán nem bontható fel. A feladó az akár nyilvánosan is közölhető szorzatot használja fel az üzenet kódolására, míg a dekódoláshoz a címzettnek a prímtényezőket is ismernie kell — az a tulajdonképpeni titok.

F. P.

Kriptográfia

A tar koponyától a becsapódó ajtóig

A titkosírás tudományának görög neve, a kriptográfia eredetileg rejtett írást is jelentett. Talán mert a rejtjelezés és az írás elrejtése akkoriban még szorosabban összetartozott. Például úgy, hogy hajának teljes lenyírása után a futár fejbőrére írták az üzenetet, és akkor indították őt útnak, amikor a haja ismét megnőtt. Persze az ókori görögök már algoritmikus eszközökkel is végeztek rejtjelezést. Ők alkalmazták először a mechanikus ráccsal való titkosítást. Egy sablonrács lyukain át, többszörös elforgatással írták le a szöveget, majd az üresen maradt helyekre megtévesztésül véletlenszerűen írtak be betűket. A megfejtéshez egy ugyanolyan lyukasztású „szita" kellett. (Dekóder.)

Másfajta algoritmikus titkosírást találtak ki a kabalisták. Nekik egy könyv, az Ószövetség héber szövege rejtette a kulcsot. A számokkal megadott oldalakon, sorokban és pozíciókon lévő betűket összeolvasva lehetett megkapni a titkolt liturgikus utalást. Ezek a rejtjeles részek sajátos CRC-ként működve biztosították a szövegek változatlan formában történő megőrzését és az utódoknak való továbbadását.

A következő lépcsőfok az úgynevezett transzlációs kódtáblák alkalmazása volt, s azok egészen az első világháború végéig használatban voltak. Ennél a titkosítási eljárásnál leírták az ABC-t, majd valamilyen megadott értékkel eltolva újra és újra leírták. A szöveg titkosításakor azután egy adott kulcs szerinti sorból vették ki a megfelelő betűt helyettesítő kódot. A megfejtéshez ismerni kellett azt az algoritmust, amelyikkel a sorokat kiválasztották, továbbá ami szerint az egyes ábácésorokat egymáshoz képest elcsúsztatták. Ez a kódolási technika azonban a számítógépek megjelenése után fokozatos kiment a divatból. Máig is rejtjeleznek viszont szövegeket kulcsként megadott könyvek betűkészletének a felhasználásával, továbbá egyszer használatos kódokkal.

Az egyszer használatos kódnál a kiindulás egy ábécé, ahol minden betű alá írnak egy (általában kétjegyű) számot. Utána ennek alapján kódolják a szöveget, és a csoportokba rendezett számsorhoz hozzáadják az egyszer használatos, megegyezéses kódlapon lévő véletlen számokat, ugyancsak megegyezéses algoritmus szerint. Ebből ún. maradékos összeadással (amikor csak az utolsó helyiértéken lévő számot veszik figyelembe) létrehoznak egy számsort, s azt mint üzenetet továbbítják. A dekódoláshoz szükség van ugyanarra a kódlapra, és tudni kell, mely számoszlopokat milyen sorrendben használtak fel a kódoláskor.

A második világháború idején kifejlesztették a németek az Enigma elnevezésű elektromechanikus kódológépet. Ennek alapján később több országban készítettek hasonló elvű berendezéseket. A matematikai kutatások eredményeinek alkalmazásával egyre nehezebben megfejthető algoritmusok születtek. Amikor azok honvédelmi alkalmazására már nem volt szükség, a titkosítási eljárások közkinccsé váltak. Igyekeztek ugyan rájuk ütni az államtitok pecsétjét, de nem nagyon sikerült.

Magyarországon az amerikai adattitkosítási szabvány (DES = Data Encryption Standard) gyorsan elterjedt. Igaz, hogy az amerikai cégek nem amerikai szervezeteknek csak olyan rendszereket adhattak el, amelyek gyors dekódolásához az NSA (National Security Agency) rendelkezett a szükséges eszközökkel, de ez a körülmény nem hátráltatta, hanem inkább elősegítette a magyar kriptológia fejlődését. Igencsak jó minőségű, nehezen megfejthető, minősített információ kódolására alkalmas saját titkosítási megoldások születtek Magyarországon.

A DES algoritmus hátránya az volt, hogy két személy közötti kommunikációhoz a kulcsot valamiképpen el kellett juttatni a partnerhez. Ez a hírlánc legsebezhetőbb pontja. A DES elég megbízható abban az esetben, ha viszonylag rövid és nem behatárolható tartalmú anyaghoz szükséges. Például a Swift elektronikus banki átutalási rendszerben az elsődleges egyéni kulcs továbbítására a DES-t használják, míg magának a DES-nek a kulcsát a kapcsolatfelvétel kezdeményezése előtt intelligens memóriakártyában továbbítják a célállomásra.

Ugyanakkor felfedezték az úgynevezett becsapódó ajtó jellegű algoritmusokat. Nevüket onnan kapták, hogy a már kódolt információ visszafejtéséhez más eljárás és más kulcs kell, mint amivel a kódolás műveleteit elvégezték. (Belülről kilinccsel nyitjuk az ajtót, de ha az becsapódik, és mi a folyosón rekedtünk, akkor már csak kulccsal — vagy lakatossal — tudunk visszamenni a lakásba.) A klasszikus, prímszámos algoritmus kissé bonyolult matematikai hátteret igényel, a mindennapi életben talán könnyebben használható a véletlenszám-generálás és a transzlációs kód bizonyos vonásait összekapcsoló megoldás. *(Ezekről szólnak írásaink a 8-13. oldalon. A szerk. megj.)*

Kis János

UNICODE



„Titok lesz, mondta Findlay."

A megfejthetetlen kód „megfejtése"

A titkosírás az elmúlt évtizedben hatalmas fejlődésnek indult, aminek fő oka, hogy erre a területre is betört a matematika. Míg korábban egy új kódrendszer bevezetésekor inkább csak a szakemberek megérzéseire támaszkodtak, hogy megállapítsák egy kód megfejtésének bonyolultsági szintjét, addig ma már természetesnek látszó követelmény, hogy a kódolás megfejthetetlenségét szigorú matematikai módszerekkel *bizonyítsák*. A magyar szellemi műhelyek ebben a témában is nemzetközi tekintélyre tettek szert. Mi az alábbi írásban és annak kiegészítő anyagaiban az MTA Matematikai Kutatóintézetének munkatársát „szólaltattuk" meg.

A matematikai tételek egyik jellemző tulajdonsága, hogy tartalmaznak feltételeket is: ha ez meg ez igaz, akkor az meg az így van. Amikor a tétel kikerül a steril matematikai környezetből, akkor igen gyakran megesik, hogy erről megfeledkeznek, a feltételek elsikkadnak, csak a következmény marad meg. Ez az oka annak, hogy néha csak úgy általában szoktak emlegetni megfejthetetlen kódokat, abszolút biztonságos titkosírást, feltörhetetlen programvédelmet.

Pedig a fentiek mind-mind csak bizonyos feltételek teljesülése mellett igazak, amely feltételek fennállásáról azonban sajnos nem tudunk semmi biztosat. Ami persze nem jelenti azt, hogy a titkosításra alkalmazott algoritmusok rosszak vagy egyenesen használhatatlanok volnának, hanem csak azt, hogy az ilyen kijelentéseket mindig fenntartással kell fogadnunk.

A lehető legnehezebb problémák

Mik is ezek a rejtett feltételek? A számítástechnikában jártasak minden bizonnyal hallottak már a $P \neq NP$ problémáról. Egészen tömören fogalmazva a kérdés az, hogy ellenőrizhető megoldással rendelkező problémáknál belátható időn belül meg tudjuk-e találni a megoldást.

Az elmélet egyik igen szép eredménye, hogy az adott problémakörben vannak lehető legnehezebb problémák, sőt a természetesen adódó feladatok széles skálája is ilyen. Ezeket NP-teljes problémának nevezik: ha ezek bármelyikére sikerülne gyors algoritmust találni, akkor az NP-beli összes többi probléma is gyorsan megoldható lenne. Ilyen feladatok például:

— Döntsük el, hogy egy n szögpontú gráfban van-e Hamilton-út (élek olyan sorozata, amely a gráf minden csúcsán pontosan egyszer megy át).

— Egy gráf csúcsai színezhetők-e három színnel úgy, hogy az éllel összekötött csúcsok különböző színűek legyenek.

— Van valahány fájl, mindegyik kisebb, mint 100 k. Minimálisan hány 360 k-s floppyra férnek fel?

— Adott számú chipet és a lábaik közötti huzalozást el kell helyeznünk egy adott méretű nyákon. Tudjuk, hogy két huzalozásnak egymástól legalább 0,2 milliméterre kell lennie. Döntsük el, megvalósítható-e a kapcsolás.

— A számítástudomány legfontosabb nyitott problémája az ún. P≠NP probléma: igazoljuk, hogy ezekre a feladatokra tényleg nincs gyors eldöntési algoritmus.

Vannak problémák, elsősorban a számelméletben, amelyek NP-beliek, de nem tudjuk, hogy NP-teljesek-e. Ilyenek a következők:

— Adott egy p szám, döntsük el, hogy p prím-e.

— Adott n szám, ami két prím szorzata. Döntsük el, hogy egy $a < n$ szám kvadratikus maradék-e, vagyis van-e olyan x, hogy $a \equiv x^2 \pmod{n}$.

— Adott egy p prím és egy g szám. Adott $a < p$-re keressünk olyan b-t, hogy $a \equiv g^b \pmod{p}$.

A legtöbb esetben (és a kódolás, a titkosírás ilyen) megelégszünk olyan algoritmusokkal, amelyek véletlenszámokkal dolgoznak, és az eredményt nem teljes bizonyossággal, hanem csak nagy valószínűséggel adják meg. Prímtesztelésre az alábbi algoritmus alkalmas:

Végezzük el a következő tesztet többször egymás után. Ha p prímszám, akkor a teszt minden esetben azt mondja, hogy „p lehet prím". Ha p nem prímszám, akkor a teszt több mint 0,25 valószínűséggel azt mondja, hogy „p nem prím". Például 50 próbálkozás után annak valószínűsége, hogy p-ről azt mondtuk, hogy prím, és mégsem az, kisebb mint 2^{-100}.

Legyen $p = (2^a)m+1$, ahol m páratlan. Válasszunk egy b véletlen értéket 1 és p–1 között, és számítsuk ki $c = b^m \pmod{p}$ értéket. Ha $c \equiv 1 \pmod{p}$ vagy $c \equiv -1 \pmod{p}$, akkor p lehet hogy prím. Ha c sem +1, sem –1, akkor c ismételt négyzetre emelésével számítsuk ki b-nek a 2m-edik, 4m-edik, 8m-edik, ..., $2^{(a-1)}$m-edik hatványait modulo p, amíg valamelyik kapott érték vagy +1 vagy –1. Ha –1 érték miatt álltunk le, akkor p lehet prím. Ha +1-re álltunk le, vagy még az utolsó tag is különbözött +1-től és –1-től, akkor p biztosan nem prím.

Megjegyezzük, hogy nem ismeretes olyan gyors (p bináris jegyeinek számában polinomiális idejű), véletlent nem használó algoritmus, amely megmondaná: p prímszám-e vagy sem. A fenti véletlen algoritmus ideje p leírásához szükséges jegyek számában polinomiális; ehhez csak a hatványozást kell ügyesen kiszámítani.

Abból, hogy NP-teljes problémákat még így, véletlenszámokat használva sem lehet nagy valószínűséggel megoldani, azonnal adódna, hogy $P \neq NP$. Következésképp *egyetlen NP-beli*

problémáról sem tudja senki bizonyítani, hogy nem oldható meg véletlen algoritmussal!

A titkosírásban, kódolásban használt algoritmusok, protokollok megfejthetetlensége mögött általában a következő feltevés húzódik meg:

Nincs olyan gyors, véletlenszámokat használó algoritmus, amely nagy valószínűséggel megoldaná az alábbi feladatokat:

— Legyen n két 4k–3 alakú prím szorzata. Döntsük el, hogy adott $a < n$ számhoz található-e olyan x szám, hogy $a \equiv x^2 \pmod{n}$ (vagyis *a* kvadratikus maradék-e).

— Adott a p prím és a $g < p$ generátor. Az adott $a < p$-hez keressünk olyan $b < p$-t, hogy $a \equiv g^b \pmod{p}$.

Az első feladat (abban az értelemben, hogy egyik megoldását ismerve a másikat is gyorsan meg tudjuk oldani) ekvivalens azzal a feladattal, hogy:

— Adott az n szám, ami két 4k+3 alakú prím, p és q szorzata. Keressük meg n ismeretében p-t és q-t.

Ez utóbbira úgy hivatkoznak, hogy „a prímfaktorizáció nehéz"; a másikra pedig úgy, hogy „a diszkrét logaritmust nehéz kiszámítani".

Még legalább 30-50 évig feltörhetetlen

A feladatok nehézségéről mindenki meg van győződve, ennek ellenére ezt senkinek sem sikerült bizonyítania. Kiterjedt kutatás folyik ezen a területen: egyrészt olyan protokollokat, kódolásokat adnak meg, amelyek akkor is garantáltan nehezen visszafejthetők, ha a fenti számelméleti problémákra mégis tudna valaki gyors algoritmust adni. Ilyenek az ún. egyirányú (trapdoor, one-way) függvények létezését használó protokollok. Másrészt a fenti számelméleti problémák megoldására új, a korábban létezőknél gyorsabb algoritmusokat keresnek. Mindkét oldalon jelentős eredmények születtek; az első esetben tetszőleges, nagyon gyenge értelemben vett egyirányú függvény létezéséből bizonyítják tetszőleges erősségű protokoll létezését (bár ilyen függvény létezéséből azonnal adódna, hogy $P \neq NP$, tehát a függvény létezésének bizonyítása ma még reménytelen).

A prímfelbontás oldalán új algoritmusok jelentek meg, s míg 15 évvel ezelőtt 30 (decimális) jegyű számok tényezőkre való bontása volt belátható időn belül (1-2 hónap futási idővel) lehetséges, ma már így 100-130 jegyű számokat tudnak tényezőkre bontani. A diszkrét logaritmus keresése is hasonló méretű prímeknél lehetséges. Ha tehát valakinek ma elég sok a pénze (kb. 1000 millió dollár a gép, illetve gépidő megvásárlására), és elég sok az ideje (kb. 3 hónap), akkor mindazokat a titkosírásokat, kódolásokat, melyek végső soron egy 100-130 decimális jegyű szám tényezőkre való felbontásának nehézségét használják ki, fel tudja törni. Hasonló idő, de ezredannyi pénz elegendő kb. 60 jegyű számok esetében, egy PC pedig 20-50 decimális jegyű számokkal tud megbirkózni. A múltbeli haladást a jövőre extrapolálva az e területen dolgozó szakemberek véleménye az, hogy 30 éven belül 512 bites kulcsokat használva a kód feltörése nagy intézményeknek (óriás bankoknak, állami apparátusnak) nem fog gondot okozni. Viszont kulcsként 1024 bites számokat használva a kódolás még 30-50 év múltán is majdnem biztosan feltörhetetlen lesz.

Csirmaz László

Digitális aláírás

1992 nyarán hozták nyilvánosságra a DSS (Digital Signature Standard) protokollt, mely digitális aláírások rögzített szabványa kíván lenni. A protokoll — hasonlóan a banki ellenőrzőhöz — digitális *aláírások* készítésére szolgál. Az üzenet tetszőleges lehet (akár kódolt, akár szimpla szöveg). Az üzenetből valamilyen *hash* függvénnyel előállítunk egy m számot, és ezt a számot kódoljuk úgy, hogy:

— Az aláírást csak az aláíró tudja elkészíteni.

— Az aláírás helyességét mindenki viszonylag gyorsan ellenőrizni tudja.

A DSS protokoll paraméterei: egy p prímszám (javaslat 1024 bites), egy q prímszám, ami osztója p–1-nek (q 512 bites) úgy, hogy a hányados páratlan szám; továbbá egy tetszőleges $g < p$, aminek (mod p) rendje éppen q. Az aláírás hamisíthatatlansága (ha van ilyen szó?) azon múlik, hogy a diszkrét logaritmust még véletlenül is nehéz kiszámítani.

Az aláíró választ saját használatra 1 és q között véletlenszerűen egy x titkos kulcsot; kiszámítja a $y = g^x \pmod{p}$ számot, és y-t nyilvánosságra hozza. Ha az aláírandó szöveg kódja m, akkor az aláírás két szám 0 és q–1 között, amit a következőképpen kell előállítani:

1. Válasszunk egy k véletlen számot 0 és q között, és számítsuk ki l-et úgy, hogy $k*l \equiv 1 \pmod{q}$.

2. Legyen $r = (g^k \pmod{p}) \pmod{q}$ (vagyis számítsuk ki g k-adik hatványát modulo p, azután az eredménynek vegyük q-val való osztásakor a maradékot).

3. Legyen $s = l*(m+x*r) \pmod{q}$ ahol x az aláíró titkos kódja, m az üzenet, r az előző lépésben kapott érték.

Az aláírás az (r,s) pár.

Az aláírás ellenőrzése a következőképpen történik: legyen az aláírás (r,s); a kódolt üzenet m, a kódoló nyilvános kódja y.

1. Ha r és s valamelyike nulla vagy nagyobb, mint q–1, akkor az aláírás rossz.

2. Számítsuk ki azt a t-t, amire $t*s \equiv 1 \pmod{q}$, legyen továbbá

$u = m*t \pmod{q}$

$v = r*t \pmod{q}$

$w = g^u * y^v \pmod{p}$

3. Ezek után ha $r \equiv w \pmod{q}$, akkor az aláírás helyes, egyébként hibás.

Az érdeklődő olvasóra hagyom annak kipróbálását, hogy helyes aláírás esetén az ellenőrzés azt el fogja fogadni.

A szabványtervezet még utalásokat tartalmaz arra nézve, hogyan tudunk megfelelő méretű prímet találni (talán ez a legnehezebb); hogyan találjunk egy q-adrendű elemet, stb. Mivel prímek találása a legnehezebb feladat, csábítónak tűnhet megfelelő méretű prímszámokat „venni". Azonban bizonyos speciális prímekre a diszkrét logaritmus feladat (viszonylag) könnyen megoldható, vagyis a prímet eladó cég esetleg tudja az aláírásunkat hamisítani. Egy véletlenszerűen választott prímszámra a jelenleg ismert algoritmusok nem működnek, és ilyen prímek használatakor az aláíró bízhat abban, hogy aláírása még legalább 50 évig nem hamisítható.

Cs. L.

A prímek viszik a prímet

Nyilvános kulcsú titkosítás

Az emberek joggal gyanakodnak: valaki mindig figyel. Figyeli, hogy mit telefonálunk, miről levelezünk. Ez a félelem hatványozottan jelentkezik azoknál, akik az elektronikus postát használják. Szemben a hivatalos állami igyekezettel — lásd az USA exportellenőrzési törvényeit — felmerült az igény, hogy a magánkommunikáció ugyanolyan titkos lehessen az idegen fülek-szemek előtt, mint a hivatalos.

A számítógépek, számítógép-hálózatok térhódításával párhuzamosan egyre növekvő jelentőségre tett szert az adatvédelem kérdése. Adatainkat a legkülönbözőbb veszélyek fenyegethetik: hardvereszközök meghibásodása, adatátviteli hibák, illetéktelen hozzáférés. Míg a hardverhibák kezelése — jellegükből következően — alapvetően új megközelítéseket igényelt, addig a hozzáférésvédelem specialistái sok esetben bőven meríthettek a tikosírások elméletének sok évszázados tapasztalataiból. A magánkommunikáció titkosításának népszerűvé és közismertté válását is az segítette elő, hogy az elméleti kutatási eredmények nyilvános publikációkban megjelentek. Így némi utánabogarászással megtalálhatóak a DES és az RSA algoritmus képletei, leírásai. Gyakorlatilag ezek a korszerű algoritmusok váltották fel a korábban alkalmazott egyedi kulcsos megoldásokat.

A történelem folyamán számtalan figyelemre méltó tikosítási módszert dolgoztak ki, közülük azonban talán a legérdekesebb a nyilvános kulcsú titkosítás (NKT), mely az utóbbi évtizedek eredménye. Hogy az NKT újszerűségét megérthessük, meg kell ismernünk a titkos üzenetváltás menetét hagyományos rejtjelezési módszer esetén.

Tegyük fel, hogy A titkos üzenetet kíván küldeni B-nek. Ekkor előzőleg meg kell állapodniuk egy rejtjelezési algoritmusban, valamint megfelelő rejtjelkulcsot kell választaniuk, amelyet rajtuk kívül senki sem ismer. A rejtjelkulcs kijelölése feltétlenül valamiféle személyes kontaktust tételez fel A és B között, azaz A-nak és B-nek ismernie kell egymást, és az üzenetváltást megelőzően a kulcsot közösen meg kell határozniuk. Ez a rejtjelezési séma tökéletesen illeszkedik a diplomácia, a hírszerzés vagy éppen a titkolódzó szerelmespárok igényeihez, a gazdasági élet szereplői azonban joggal elégedetlenek lehetnek vele.

Tegyük fel ugyanis, hogy A egy nagy példányszámú napilapban meghirdeti, hogy eladni szándékozik bizonyos ingatlanjait. B — aki előzőleg soha nem találkozott A-val — minél előbb reagálni akar a hirdetésre árajánlatának eljuttatásával. Ajánlatát azonban kódolva szeretné A-hoz küldeni, oly módon, hogy a címzetten kívül senki más ne tudja elolvasni. Mi a teendő?

A megoldás a nyilvános kulcsú titkosítás. A módszer alapgondolata a következő: olyan partnerek között is lehetővé kell tenni a rejtjelezett kommunikációt, akik előzőleg nem cseréltek titkos kulcsot egymással.

Ha A be akar lépni a nyilvános kulcsú titkosítók közösségébe, akkor megfelelő módon két kulcsot kell generálnia: S_A-t és P_A-t.

P_A A nyilvános kulcsa, ezt mindenki számára közzé teszi.

S_A A titkos kulcsa, rajta kívül más nem ismeri.

A nyilvános kulcsok közzététele például telefonkönyvszerű formában történhet, vagy számítógépes rendszer esetén a kulcsok a hálózat egyik gépén, bárki által hozzáférhető adatbázisban tárolhatók.

Ezt követően, ha B titkos üzenetet kíván eljuttatni A-hoz, akkor kikeresi P_A-t, és az M üzenetet ennek felhasználásával kódolja.

A rejtjelező algoritmus nyilvános, mindenki ismeri. A kódolt üzenetet A saját, mások által ismeretlen S_A titkos kulcsával fejti vissza.

A rejtjelezési séma formálisan a következőképpen írható fel:

1. lépés: $E = C_A(M)$ (B végzi)

C_A Kódoló algoritmus A nyilvános kulcsának felhasználásával

E A kódolt üzenet

2. lépés: B – (E) –> A A rejteles üzenet elküldése

3. lépés: $M = D_A(E)$ (A végzi)

D_A Dekódoló algoritmus A titkos kulcsának felhasználásával

A módszer megvalósíthatósága azon múlik, lehetséges-e olyan kódoló/dekódoló algoritmust találni, mely kielégíti az alábbi követelményeket:

1. A rejtjelezés a nyilvános kulcs ismeretében egyszerűen végrehajtható legyen.

2. A visszafejtés csak a titkos kulcs ismeretében legyen egyszerű, egyébként nehéz vagy lehetetlen.

A nyilvános kulcsú titkosítás fenti alapelveit Markle, Diffie és Hellman publikálta 1976-ban. Cikküket sokan kétkedéssel fogadták, kétségbe vonva a megfelelő algoritmus létezését. A megvalósításra azonban csak két évet kellett várni. 1978-ban Rivest, Shamir és Adleman nyilvánosságra hozta a róluk elnevezett RSA algoritmust. A módszer megértéséhez szükséges a matematika kongruencia fogalmának tisztázása.

Két nemnegatív egész szám egy rögzített m (modulus) pozitív számra nézve akkor kongruens egymással, ha m-mel osztva azonos maradékot adnak.

Jelölés: $a \equiv b \ (m)$, a kongruens b-vel m-re nézve

Például $5 \equiv 8 \ (3)$, mivel 3-mal osztva 5 és 8 maradéka is 2

$4 \equiv 14 \ (5)$

A kulcsválasztás ennek alapján a következőképpen történik:

Választanunk kell először két igen nagy p és q prímszámot. Később részletezendő okok miatt több mint 100 jegyű prímeket kell keresnünk. A prímek felhasználásával kiszámítjuk az $m = pq$ modulust és az $F = (p-1)(q-1)$ értéket, majd keresünk egy olyan r számot, melynek legnagyobb közös osztója (p-1)-gyel és (q-1)-gyel egyaránt 1 (relatív prímek).

Végül választunk egy olyan k számot, mellyel az rk szorzat F-fel osztva 1 maradékot ad. Ezt követően az m,r

számpárt nyilvános kulcsként közzétesszük, a k, p, q számhármast titkos kulcsként őrizzük meg.

A rejtjelezés folyamata RSA algoritmus szerint B –> A üzenetküldés esetén:

Karakteres üzenetét B-nek először nemnegatív egész számok sorozatává kell alakítania. Számítógép esetén az átalakítást adottnak vehetjük, ui. az üzenet a gépen — általában — ASCII kódok (számok) formájában tárolódik. Az átalakítás természetesen tetszőleges egyéb módszer szerint is történhet, lényeg az, hogy valamennyi nyilvános kulcsú titkosító ismerje a konverziós szabályokat. Az átalakítást követően B kikeresi A nyilvános kulcsát, vagyis az r_A, m_A számpárt. A kódolást a számsorozattá alakított üzenet egyes értékein egymás után, az alábbi módon végzi el: legyen a következő kódolandó szám n. A neki megfelelő e rejtjelezett szám az $e \equiv n^{r_A}$ (m_A) összefüggéssel adódik. Azaz elvégezzük az n^{r_A} hatványozást, majd az eredménynek képezzük m_A-val osztva a maradékát. Ez lesz az e.

A dekódolás folyamata:

A megkapja a B által küldött rejtjelezett számsorozatot. A sorozat elemeit külön-külön, azonos módon dekódolja. Legyen a soron következő dekódolandó szám az e. Az e-nek megfelelő dekódolt n szám az $n \equiv e^{k_A}$ (m_A) összefüggéssel áll elő. Azaz e-t a k_A-dik hatványra emeljük (k_A-t csak A ismeri!), és képezzük m_A-val osztva a maradékát. Ez a maradék lesz n, az eredeti üzenet megfelelő száma! A módszer működőképessége a számelmélet egy nevezetes tételén, a kis Fermat-tételen alapszik. Ui. $e^{k_A} = (n^{r_A})^{k_A} = n^{r_A k_A}$, tudjuk továbbá, hogy $r_A k_A \equiv 1$ (F), hiszen így választottuk. A Fermat-tétel garantálja, hogy ilyen feltételek esetén $n^{r_A k_A} \equiv n$ (m_A). A dekódolt számsorozatot az ismert konverziós szabállyal alakítja vissza A karakterekké.

A módszer kétségkívül működik, kérdés azonban, hogy kielégíti-e a korábban felsorolt követelményeket.

1. A kódolás, egyszerű modulo m hatványozásról lévén szó, a nyilvános kulcs ismeretében, számítógéppel könnyen megoldható. Mindössze olyan szoftverre van szükség, amely képes többszáz jegyű egész számok kezelésére.

2. A dekódolás ugyancsak hatványozással történik, mely számítógéppel, a titkos kulcs ismeretében gyorsan elvégezhető.

3. A titkos kulcs ismerete nélkül a dekódolás gyakorlatilag megoldhatatlan, mivel ez a nyilvánosságra hozott m_A modulus prímtényezőkre bontását (p és q meghatározását) igényelné. Mai ismereteink szerint nem létezik olyan algoritmus, mellyel ezt a feladatot belátható időn belül el lehetne végezni. Viszonyításképpen: az ismert algoritmusokkal tetszőleges, több mint 200 jegyű szám prímtényezőkre bontása, figyelembe véve a jelenlegi számítógépek teljesítményét, évezredekig tartana. A felbontás tehát nem elvileg, hanem csak gyakorlatilag lehetetlen! Ennek értelmében a számítógépek teljesítményének növekedésével párhuzamosan egyre nagyobb és nagyobb prímeket kell használnunk!

4. Lényeges kérdés, hogy a kulcsválasztás során, elfogadható időn belül tudunk-e nagy prímeket generálni.

A prímgenerálás tehát a következőképpen történik:

— Előállítunk egy elég nagy páratlan véletlenszámot.

— Prímteszt alkalmazásával megállapítjuk, hogy prím-e vagy összetett.

— Ha összetett szám, akkor újabb véletlenszámmal próbálkozunk.

Talán meglepően hangzik, de tény, hogy kb. 100 jegyű számok esetén átlagosan kevesebb mint másfélszáz próbálkozás eredményhez szokott vezetni.

Az RSA módszer természetesen nem az egyetlen nyilvános kulcsú titkosítási algoritmus. Az elmúlt két évtizedben igen nagy erőfeszítéseket tettek ilyen jellegű technikák kidolgozására. A publikált algoritmusok többsége azonban feltörhetőnek bizonyult. Feltétlenül említést érdemel azonban a LUC eljárás, mely az RSA érdekes alternatíváját jelentheti.

Végezetül pedig néhány megjegyzés a nyilvános kulcsú titkosítás gyakorlati megvalósításával kapcsolatban: tény ugyanis, hogy az RSA algoritmus ma már jóval több, mint holmi matematikai érdekesség.

1983. szeptember 20-án 4 405 829-es számmal szabadalmaztatták az RSA algoritmust, és 1984-ben megalakult az RSA Data Security Inc. a módszer hasznosítására. Azóta a Lotus és a DEC már felhasználta az eljárást bizonyos rendszereiben hitelesítési és titkosítási célokra, az RSA része a Netware operációs rendszernek, s a Sun, a Microsoft és az IBM is megvásárolta a módszert későbbi fejlesztések céljára.

A magánkommunikáció terén az egész világon — az amerikai dörgedelmek ellenére — szabványossá vált az utóbbi idők egyik legkomolyabb titkosító eljárása, a PGP. Ez is az RSA titkosítási algoritmust alkalmazza. Biztonsága attól függ, milyen kódhosszúságú kulcsokat használnak. A közforgalomban lévő kódoló program 288, 512 valamint 992 bájt hosszúságú kulcsokat alkalmaz. Az utóbbi megegyezik a külhoni gyakorlatban a minősített információk továbbítására alkalmazott kulcshosszal.

A PGP a fent leírt lépéseket automatikusan menedzseli. A titkosítás során a kiinduló normál ASCII állományból a két kulcs segítségével előállít egy titkosított RSA-Enziphered szövegállományt, amely látszólag véletlenszerűen ALT 127-ig terjedő karakterekből álló massza.

A program lehetőséget ad az egyedi és titkos kulcsok generálására. A titkos kulcsot egy transzlációs eljárással készült önálló titkosítási rutin teszi továbbíthatóvá. A nyilvános, illetve titkos kulcsok állományokban külön elküldhetők az üzenet fogadójához akár másmás útvonalon is.

Az RSA algoritmus elterjedése forradalmasította a privát kommunikációt. Minden hatósági tiltás ellenére használata terjedőben van. Az UUCP és újabban a FIDO levelezési rendszeren elterjedten használt algoritmus. Amit az sem tud megakadályozni, hogy az USA-ban az ilyen szoftverek terjesztését és nem állami alkalmazását kifejezetten tiltják, magán- és vállalati célokra csak viszonylag könnyen megfejthető eljárásokat szeretnének alkalmaztatni. A privacy-ért folytatott harc viszont — úgy tűnik — újabb területen kezd csatát nyerni. Mindenkinek joga van számítógépében tárolt adatait és saját kereskedelmi titkait megvédeni az általa illetékteleneknek nyilvánítottaktól. Az újabb polgárjogi mozgalmak már ezt is zászlajukra tűzték.

Kis János—Nagy Sándor

Ajánlott szakirodalom:

Stewart: A matematika problémái, Akadémiai Kiadó, 1991.

Nemetz-Vajda: Algoritmusos adatvédelem, Akadémiai Kiadó, 1991.

Schneier: Untangling Public-Key Cryptography (Dr. Dobb's Journal, 1992/5).

Smith: LUC Public-Key Encryption (Dr. Dobb's Journal, 1993/1).

A Hamming-algoritmus

Hibajavító kódolás

Adatok elektronikus úton történő tárolása, illetve továbbítása szükségszerűen együtt jár a különböző átviteli hibák megjelenésével. Mivel a hibák mennyisége fizikai eszközökkel csak csökkenthető, de teljesen nem küszöbölhető ki, szükségessé vált olyan kódolási technikák kifejlesztése, amelyekkel jelezhetők vagy akár ki is javíthatók az információtovábbítás során fellépő rendellenességek.

Hibavédelem céljából az átvinni kívánt adatokhoz — megfelelő módon — további járulékos információkat kell csatolni. A csatolt redundáns bitek értéke szükségszerűen az adatbitek függvénye. A függvény alkalmas megválasztásával hatékony hibakezelő eszköz kerülhet a birtokunkba.

A továbbiakban nevezzük blokknak az adat és a redundáns információ összességét. Ha d az adatbitek száma, r pedig a redundáns bitek mennyisége, akkor a blokk hosszúsága n = d + r. Az így definiált blokkot n elemű vagy n bites kódszónak is nevezik.

Ennek megfelelően hibajavító kódolásnak nevezzük azt az esetet, amikor a kódszó annyi redundanciát tartalmaz, hogy annak felhasználásával a vevő hiba esetén az eredeti információt rekonstruálni képes. Hibajelző kód esetén a redundancia csak a hiba tényének detektálásához elegendő, kijavításához már nem.

Az n bites blokkok esetén a lehetséges kódszavak száma 2^n. A redundáns bitek kiszámításának módját úgy szokás megválasztani, hogy a lehetséges blokkoknak csak egy része legyen legális. Legálisak azok a kódszavak, amelyek a normális — értsd hibamentes — adatátvitel során előfordulhatnak. A módszer nyilvánvaló következménye, hogy ha a vevő illegális kódszót kap, akkor bizonyos lehet benne, hogy adatátviteli hibával van dolga.

Két kódszó Hamming-távolságán azoknak a bitpozícióknak a számát értjük, melyeken a megfelelő bitek értéke egymástól különböző. Például 00110110 és 10110001 kódszavak Hamming-távolsága 4. Általánosságban, ha két kódszó Hamming-távolsága e, akkor e darab bithibával alakulhat át egyik a másikba.

Kódszavak Hamming-távolságának felhasználásával a teljes kód Hamming-távolságát is értelmezhetjük: vegyük az összes legális kódszót, határozzuk meg páronként a Hamming-távolságokat, s a kapott értékek minimuma adja a teljes kód Hamming-távolságát, vagyis a bithibáknak azt a számát, amennyivel már egy legális kódszó egy másik legális kódszóba átalakulhat.

A Hamming-távolság fogalmának felhasználásával számszerűsíthetővé válik egy tetszőleges kódolási forma hibajavító vagy hibajelző tulajdonsága.

A hibajavító kód olyan kód, amelynek Hamming-távolsága e+1 (e=1,2,3...). Az e+1 Hamming-távolság esetén legfeljebb e darab bithiba bekövetkezése észlelhető, ui. ennyi hiba még biztosan illegális kódszót eredményez. Ha a hibák száma e-t meghaladja, megtörténhet, hogy újabb legális blokk keletkezik.



— Ebbe a programba soha nem tudunk belépni! Spéter Erzsébet születési dátuma a jelszó...

A hibajelző kód olyan kód, melynek Hamming-távolsága 2e+1 alakban írható fel (e=1,2,3...). Az e+1 Hamming-távolság esetén legfeljebb e darab bithiba javítható, ui. ekkor még a keletkezett illegális blokkhoz a legális eredeti kódszó van a legközelebb. Ha a bithibák száma e-nél nagyobb, akkor a javítás során esetleg helytelen döntésre juthatunk.

A legegyszerűbb hibajelző kód a sokak által jól ismert paritásbit módszer. Lényege, hogy az adatbitekhez 1 darab paritásbitet fűzünk, melynek értékét úgy kell megválasztani, hogy például az 1-es értékű bitek száma mindig páros legyen.

Példa

0111011 1 legális (6 db 1-es bit van)
0110000 0 legális (2 db 1-es bit van)
0110101 1 illegális (5 db 1-es bit van)

A kód Hamming-távolsága 2, azaz 1 bithiba észlelésére elegendő, 2 már becsaphatja.

A gyakorlatban hibajelzésre általában CRC kódolást használnak. Itt a paritásbitek kiszámítása polinomosztással történik, aminek eredményeként több független bithiba, illetve csoportos hibák kimutatása is lehetővé válik.

A legegyszerűbb hibajavító kódolási algoritmus az alap paritásbit módszer továbbfejlesztésével adódik: az adatokat táblázatba rendezzük, majd a sorokra és oszlopokra egyaránt paritásbitet számolunk.

Példa

0110110 0
1011001 0
1111100 1
0010011

Bithiba esetén a megfelelő sor és oszlop paritásbitjei nem megfelelőek, így automatikusan kiadódnak a javítandó bit koordinátái. A módszerrel 1 hibát javíthatunk, 2-t pedig jelezhetünk. Tegyük fel ugyanis, hogy ugyanabban a sorban 2 hibás bit fordul elő. Ekkor a sor paritásbitje jó, ellentétben a hibás oszlopok ellenőrző bitjeivel. Tudjuk tehát, hogy a hibás bitek mely oszlopokban találhatók, de azt nem, hogy melyik sorban.

A gyakorlatban a hibajavító kódolás legelterjedtebb módszere a Hamming-algoritmus. Lényege a következő:

— Számozzuk meg a kódszó bitjeit 1-től kezdődően. Azok a bitek lesznek a paritásbitek, amelyek sorszáma 2-hatvány, a többi bit adatbit.

Az ellenőrző bitek tehát az adatbitek közé keverednek.

— Az ellenőrző bitek értékét mindig úgy kell megválasztani, hogy a hozzá tartozó bitcsoport paritását például párossá tegye. Hogy valamely adatbit mely ellenőrző bitekhez tartozik, úgy dönthetjük el, hogy az adatbit sorszámát 2-hatványok összegeként állítjuk elő.
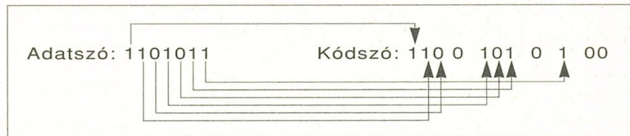
Példa

A kódszó 7. sorszámú bitje adatbit, mivel a 7 nem 2-hatvány. 7 = 4 + 2 + 1, tehát a 7. bit a 4. a 2. és az 1. sorszámú ellenőrző bitek csoportjába is beletartozik. Egy adatbitet tehát több paritásbit is felügyel.

Példa

A kódolás programozása rendkívül egyszerű. Az adatbiteket egymás után írjuk be a kódszóba. Minden egyes beírást követően (a beírt értékkel összhangban) beállítjuk a megfelelő paritásbiteket (ha a beírt érték 1, a hozzá tartozó paritásbiteket invertáljuk, egyébként változatlanul hagyjuk). Miután az adatszó utolsó bitjét is bevittük a kódszóba, az ellenőrző bitek az adatokkal automatikusan összhangban lesznek.

— Hibajavítás: Ha például a 7. bit (adatbit) értéke meghibásodik, akkor az 1., a 2. és a 4. sorszámú paritásbit értéke helytelenné válik. Ha tehát a hibás paritásbitek sorszámát összeadjuk, abból automatikusan adódik a javítandó bit pozíciója. Ha pedig az átvitel során paritásbit sérül meg, annak nincs hatása az adatokra.



A dekódolás szintén könnyen programozható. A kódszóból sorban egymás után kiolvassuk az adatbiteket, a kiolvasott bitpozíció értékét töröljük, s a hozzá tartozó paritásbiteket ennek megfelelően állítjuk be (ha a kiolvasott érték 1 volt, a hozzá tartozó paritásbiteket invertáljuk, egyébként változatlanul hagyjuk). A kiolvasás végeztével a kódszó értéke vagy nulla (nem történt hiba), vagy binárisan annak a bitnek a sorszámát tartalmazza, amely megsérült az átvitel során.

Az alap Hamming-kód blokkonként csak egyedi hibák javítására alkalmas. Az adatoknak az előző példához hasonló táblázatba foglalásával azonban már csoportos hibák is kezelhetővé válnak. A lényeg itt is az, hogy a sorokat és oszlopokat egyaránt Hamming-kódolni kell.

A Hamming-kódolás menetét lépésenként végigkövethetjük a lemezmellékleten található HDEMO.EXE program felhasználásával. A program bitenként kéri be az adatszót, s minden egyes bit beolvasását követően megjeleníti a kódszó aktuális értékét. A beolvasás végeztével lehetőségünk van bithibát vinni a kódszóba, majd lépésenként — bitenként — követhetjük a dekódolás folyamatát. Hiba esetén jól láthatóan adódik a javítandó (invertálandó) pozíció sorszáma.

A lemezmelléklet másik programjával (HCODE.EXE) tetszőleges lemezes fájlt hibajavító kódolással archiválhatunk. A program különösen akkor hasznos, ha a fájlokat hosszabb időre szeretnénk megőrizni, megóvni a hőmozgás következtében bekövetkező bithibáktól.

Nagy Sándor

A bank és a dugó

A kódok világában méltán nevezetes a Feige–Fiat–Shamir protokoll. Mondjuk, egy bank hitelkártyákkal látja el ügyfeleit. Azt szeretnék, ha az ügyfél (illetve a kártyája) mindig tudná magát igazolni, akár telefonvonalon keresztül is; de az, aki az ellenőrzést elvégzi, később ne tudja „megszemélyesíteni” a korábban ellenőrzöttet, vagyis ne tudjon úgy viselkedni, mintha a kártya a birtokában volna.

Ugyanez a protokoll alkalmas az ún. „dugós” hardvervédelemre is. Az ilyen védelem fő gyengéje nem az, hogy a védett program „visszafejthető”, hanem hogy a program és a „dugó” közti kommunikációt el lehet tárolni (386-os gépeken virtuális üzemmódban adott porthivatkozások esetén interruptot lehet kérni). Ha legközelebb a program a „dugóval” akar beszélgetni, egyszerűen azokat a válaszokat „hazudjuk” a programnak, amiket hallani akar. A Feige–Fiat–Shamir protokoll ez ellen is véd: jó pár kommunikáció ismerete sem nyújt semmi támpontot arra, hogy a program következő kérdésére mit kell válaszolni.

Legyen tehát A a kártyatulajdonos (vagy a „dugó”), B pedig az, aki a kártyát ellenőrizni akarja. Választunk két 4k+3 alakú p és q prímszámot, p-t és q-t titokban tartjuk és nyilvánosságra hozzuk szorzatukat, n = p*q-t.

Választunk s1,s2,...,sk számokat 1 és n között véletlenszerűen. Ezeket csak a kártya fogja ismerni, a számok sehol máshol nem szerepelnek. Kiszámoljuk a v1,v2,...vk számokat úgy, hogy $sj^2 * vj \equiv 1 \pmod{n}$ legyen. (Ha sj n-hez relatív prím, akkor ilyen vj létezik, és gyorsan ki is számítható például az euklidészi algoritmus alapján.) A v1,v2,...,vk értékek a *nyilvános kulcsok*, ezeket mindenki ismerheti, például a kártyára rá is lehetnek nyomtatva. A kártya azzal igazolja magát, hogy meggyőzi az ellenőrzőt, mondván, ismeri az s1,s2,...,sk titkos kulcsokat. Itt (is) használjuk azt a feltevést, hogy nem tudunk gyorsan négyzetgyököt vonni; vagyis az sj értékeket senki sem tudja a vj-k alapján kiszámítani.

Az ellenőrzés során a következő párbeszéd zajlik le A és B között t-szer, ahol t a protokoll egyik biztonsági paramétere:

1. lépés: A választ egy r véletlen számot 1 és n között, kiszámítja az $x \equiv r^2 \pmod{n}$ értéket, és x-et elküldi B-nek.

2. lépés: B választ e1,e2,...,ek biteket, ahol k a kulcsok száma, és elküldi ezeket A-nak.

3. lépés: A kiszámítja az $y = r * s1^{e1} * s2^{e2} * \ldots * sk^{ek}$ szorzatot, és elküldi B-nek.

4. lépés: B ellenőrzi, hogy $x = y^2 * v1^{e1} * v2^{e2} * \ldots * vk^{ek}$.

Ha ezek egyenlők, akkor ez a párbeszéd sikeresen fejeződik be. Ha nem egyenlők, akkor A be akarta csapni B-t, és a folyamat befejeződik. Kimutatható, hogy ha valaki (akár több) párbeszéd megfigyelése után sikerrel be tudja csapni B-t, akkor nagy valószínűséggel fel tudja bontani n-et prímtényezők szorzatára is. Feltevésünk viszont, hogy ezt nem lehet megtenni.

Csirmaz László

Fejlődéstörténeti morfondírozások

A kettes számrendszer nem végállomás

A számítástechnika fejlődéstörténetét elég mostohán kezeli a tudomány. Egyrészt kevesen foglalkoznak vele, másrészt pedig akik művelik, meglehetősen egyoldalúan teszik. A számítástechnika szédítően gyors fejlődése tudományos szempontból nem egyértelműen diadalmenet: vannak például olyan fontos kérdések, amelyek — több mint harminc éve — megoldatlanok, és egyre kevésbé akarnak foglalkozni velük azok, akiktől pedig ez joggal elvárható lenne.

A számítástechnika fejlődéstörténetének kutatása manapság kizárólag a hardver, a szoftver és némiképp az alkalmazások területére korlátozódik. Pedig sok tanulsággal járna a számítástechnika fejlődését a gépekkel foglalkozó ember szempontjából is alaposabb vizsgálatnak alávetni. Nem állítjuk persze, hogy ezt a munkát minden esetben kényelmesen el is lehet végezni. Olyannyira nem, hogy még az is könnyen megeshet, hogy egyes, a számítástechnika fejlődése szempontjából fontos, napjainkban zajló folyamatokról sohasem lesz elég információja a technikatörténésznek, így ezek a területek örökre elvesznek a történelemkutatás számára. Talán ilyen lesz a szoftvergyártás munkaszervezésének fejlődése is.

A programozás metamorfózisa

A hardverfejlesztés históriája is inkább lesz töredékes, mint teljes, különösen, ha az alkatrészfejlesztést is ebbe a körbe számítjuk. A számítástechnikával azonban nemcsak a hardver vagy a szoftverfejlesztés oldaláról foglalkozik ember, hanem a mindennapi tömeges felhasználást is ember végzi. Ennek a történetét minden bizonnyal könnyen meg lehet majd írni. Annak a felhasználói (és felhasználási) körnek a tevékenységét sem lesz nehéz történelmi szempontokból feldolgozni, amelynek mindennapi munkája többé-kevésbé egyedi programok megírásából áll. Ezek a programozók hatalmas tömeget tesznek ki.

Az egyedi programkészítés szempontjából az ember-gép kapcsolatnak három jól elkülöníthető szakasza van. Kezdetben a programozást diplomások végezték. Főleg matematikusok és villamosmérnökök. Ezek a „programozók" azonban nemcsak a programokat írták, hanem maguk ültek a gép előtt is, és személyesen végezték a programok próbáit, sőt a programfuttatást is. (Hazánkban ez az „ősgépek" időszakában volt jellemző. Így volt pl. az M-3, az Ural-1, az Ural-2 környezetében, és még kicsit később is, a különleges esetekben, és az olyan „gépkülönlegességeknél" mint pl. az UMC-1.) Ez ma érthetetlenül hangzik. Mi lehetett az ok, ami miatt matematikusoknak és mérnököknek kellett a gépek elé ülni, és még a programfuttatást is nekik kellett végezniük?

Legnyomósabb okként azt említhetjük, hogy sokkal bonyolultabb volt az átlagos programok elkészítésének módja, mint ma. Akkoriban — mivel nem volt más lehetőség — gépi kódban kellett programozni, és ehhez ismerni kellett a gép teljes hardverfelépítését, az utasítások szerkezetét és végrehajtásának módját is. Természetesen mindent bináris (esetenként oktális) formában kellett írni és kezelni. A programok áttekintéséhez, a programfutás alatti események értelmezéséhez, a sok tényleges és lehetséges bonyodalom közötti eligazodáshoz valóban nem volt elég a középfokú képzettség.

A programozók akkori teljes kiszolgálatlanságára jellemző, hogy például az Ural-1 gép közvetlenül csak „fixpontos" műveleteket tudott végezni. A gépet már jó ideje használták, mikor az ország akkori egyik legismertebb számítástechnikus matematikusa megírta és közreadta a „lebegőpontos" összeadás, szorzás és osztás programját. Ezek a programok az akkori programozótársadalomban közfigyelmet keltettek, mindenki örömmel vette használatba őket. (A hatvanas évek első felének egyik számítástechnikai „ősfolyóirata", a Központi Statisztikai Hivatal Elektronikus Számológép Részlegének közleménye, a „Gépek és Programok" jelentette meg ezeket az akkoriban szenzációsnak számító programokat.)

A második fejlődési szakasz a nagy számítóközpontok kialakulásának és felfutásának időszakával esik egybe. Ekkor már — elhanyagolható kivételtől eltekintve — magasabb szintű nyelveken folyt a programozás, és a programozó (szintén elhanyagolható kivételektől eltekintve) be sem tehette a lábát a gépterembe, nemhogy maga végezte volna programjának futtatását. Ebben az időben a sikeres programíráshoz általában már nem volt nélkülözhetetlen a hardver felépítésének ismerete, és az sem, hogy az utasítások szerkezetét, végrehajtásának módját a programozó ismerje.

A harmadik korszak a személyi számítógépek tömeges elterjedésének időszaka, amelyben élünk. A nagy számítóközpontok ugyan változatlanul működnek, a programozói munkának és a programfuttatásoknak a túlnyomó többségét azonban újra a gép előtt ülő emberek végzik. Csakhogy ezek a gépek már lakások kényelmes sarkaiban vagy hivatali íróasztalokon vannak. Kezelésükhöz nem kell villamosmérnöki vagy matematikusi diploma, és a kiszolgáltság foka olyan, hogy az már szinte kényelmetlen. Akik ma leülnek a képernyő elé, és sikeresen programoznak, azoknak csak töredéke van tisztában a gép felépítésével, és még annál is kevesebben tudják pl. azt, hogy a gép hogyan hajt végre egy kétszeres pontosságú osztást.

Mit nem kell tudni a programozáshoz?

Kezdetben az volt a kérdés, hogy mit kell tudni a programozáshoz. Mindent tudni kellett, amit a gép nem tudott, és

még azt is kellett tudni, hogy amit a gép tud, azt hogyan csinálja. A kezdeti időszak után a figyelem egyre jobban elfordult (mert elfordulhatott!) a gép felépítésétől és az utasítások végrehajtásának módjától.

Ma már a személyi számítógépeket csupán használó tömegeknek nincs is gép-architektúra fogalmuk, a magasabb szintű nyelveken programozók alig ismerik a mikroprocesszorok felépítését, a számok és utasítások gépi reprezentációját és az utasítások végrehajtásának módját. Egyszerűen azért, mert erre általában már nincs is szükség.

Ma már azt a kérdést, hogy mi mindent kell tudni a programozáshoz, kifejezőbb úgy feltenni, hogy mi mindent nem kell tudni a programozáshoz, hiszen a gép nagymértékben kiszolgálja a programozókat.

A programozók ki vannak szolgálva, és ez a kiszolgálás napról napra többet nyújt. Ez való igaz, de állítható-e az is, hogy a kiszolgálás fejlődése egyenletes, és minden lényeges területre kiterjed? Aligha. A mai kényelem egyáltalán nem jelenti azt, hogy azok a problémák, amelyekkel a gépkonstruktőrök és a programozók a kezdeti időkben viaskodtak, és a programozóknak sok kényelmetlenséget okoztak, mára már minden szempontból meg is lennének oldva.

Számrendszer a szőnyeg alatt

Vegyük például a számrendszer kérdését! Forgalomban van egy — enyhén szólva kétes értékű — tudományos tétel, mely szerint a kettes alapú számrendszer „optimális". Nem sokat ér sem a tétel, sem a „bizonyítása". Ettől függetlenül, a gépekben jelenleg is a kettes számrendszert használják, mégpedig kizárólagosan, és magától értetődő természetességgel.

Az a hajdan élénken kutatott kérdés, hogy mi az információ gépen belüli reprezentálásának „optimális" módja, hallgatólagos közmegegyezéssel a szőnyeg alá söpörtetett. Az információ gépen belüli reprezentálási módja körüli lényegi tisztázatlanságok az alapokban lévő tisztázatlanságok.

Teljesen érthető, de nem megbocsátható, hogy a gyártók azt igyekeznek propagálni, amiben jók, sőt jobbak, mint konkurenseik, és mindegyik hallgat arról, amiben gyenge, és még inkább hallgat arról, amiben úgy gyenge, hogy arról egy szót sem tud mondani. És különösen nyomósak e hallgatás üzleti okai akkor, amikor ez a kínos állapot az ünnepelt és sikeres gyártmányok elvi, konstrukciós alapjaival kapcsolatos.

Korunk számítástechnikájának általánosan gyenge pontja az információreprezentáció kérdése (minden szinten és minden téren), és az ezzel összefüggő kérdés, a numerikus műveletek végrehajtási algoritmusainak feltáratlansága. A gyártók üzleti érdekeinek szempontjából érthető, hogy ezek köré a szakmai szempontból presztízsrontó kérdések köré miért sűrűsödött közmegegyezéssel olyan mély hallgatás. Csak-

hogy ennek a hallgatásnak nagy ára lesz, mert az üzleti érdek esetünkben nem azonos a tudományos érdekkel, amely viszont most — kivételesen —, a közönséges gyakorlati érdekekkel is egybeesik.

Az utasítások finomszerkezete

Vegyük a már említett példát! Hogyan hajt végre egy kétszeres pontosságú osztást az a gép, amellyel éppen dolgozunk? És mi ennek a műveletnek a pontossága? Egyáltalán milyen módokon lehet, és mikor milyen módokon célszerű értelmezni a pontosságot ilyesféle esetekben? (Kérdezzük az olvasótól, hogy hány olyan hazai vagy külföldi céget tud megnevezni a személyi gépek gyártói közül, amelyik közzétette volna, hogy gépe milyen pontossággal végzi az összeadást, szorzást, osztást, hatványozást, és adataik meg is felelnek a valóságnak?)

A pontossági kérdésekre alig van programozó a Föld hátán, aki kielégítő választ tud adni. A válasz azonban elvileg és gyakorlatilag is megadható, ha nem is éppen könnyen. Sok utánajárást vagy sok személyes kutatást igényel. Dehát miért is érdekesek ma egy művelet végrehajtásának elemi lépései? Ki és miért lenne kíváncsi egy utasítás finomszerkezetére? Eddig még senki nem reklamált ilyen hiba miatt. Ez igaz, de az is igaz, hogy eddig még senki sem bizonyította be, hogy semmilyen szempontból nincs jobb mód a szóbanforgó absztrakt algebrai műveletek gépi közelítésére, mint ami ma használatos. Senki nem panaszkodott, hogy azért kapott volna hibás eredményt, mert gépe kettes alapú számrendszerben dolgozik. Tagadhatatlanul így van. De az is tagadhatatlan, hogy nemcsak kettes alapú számrendszerben lehet megvalósítani a számok gépi reprezentációját, és ezeknek a lézető más módoknak az összehasonlítása mind a mai napig nem történt meg.

Olyan egyszerű kérdésekre sem tudjuk a választ például, hogy csak 0 és 1 jegyekből álló bináris sorozatokkal dolgozva milyen interpretációs lehetőségeink vannak, és mikor melyiket érdemes használni. Nyilvánvaló ugyanis, hogy a csak 0 és 1 jegyeket tartalmazó sorozatokat is sokféleképpen lehet interpretálni, nemcsak úgy, mint kettes alapú számrendszerben felírt számokat, hanem pl. úgy is, mint –2 (azaz mínusz kettő) alapú számrendszerben felírt számokat. Mondhatjuk erre, hogy „jó, jó, de minek?". A válasz talán meglepő: ezt is meg kell vizsgálni. Gondosan meg kell vizsgálni, hogy van-e egyáltalán valamilyen előnye a csak 0 és 1 jegyekből álló bináris sorozatok ma szokásostól eltérő interpretációjának? Véleményem szerint van, pl. a sebesség terén.

Gyakorlati szempontból támasztja alá ezt az álláspontot az a tény, hogy gyártottak már sorozatban olyan gépet is, amely mínusz kettes alapú számábrázolást használt. Ilyen gép a hatvanas évek közepén Magyarországon is működött. Lengyelországban gyártották, UMC-1 néven hozták forgalomba, hosszú élettartamú nyugati csövekből építették. Még a számítástechnika-történészek közül is kevesen tudnak hajdani létezéséről. Akik tudnak róla, fejlődési zsákutcaként, sikertelen kísérletként tartják számon. Véleményünk szerint ez az ítélet elhamarkodott. Nagyon valószínű, hogy rövidesen egy jelentős, új fejlődési irány egyik elemének korai megjelenését fogjuk majd felismerni benne.

Pogány Csaba

A többnyelvűség

Programozói berkekben gyakran hallható az a bosszús megjegyzés, hogy „ezt a feladatot igen nehézkes megoldani ezen a programnyelven", vagy „ilyen apró problémával hetekig kínlódni, nem embernek való". Mindez arra utal, hogy a programozónak elvi bajai vannak a munkaeszközével — magyarán utál úgy kódolni, ahogy a kényszerűség diktálja. Szenved, mert nem dolgozhat a saját feje és stílusa szerint, pedig ő azt szeretné, hogy...

Máskor nem a feladattal, hanem a nyelvvel van baj. „Bezzeg X.Y.-éknál Z. programnyelvet használnak, milyen jó nekik". Persze ha X.Y.-t is megkérdezzük, az meg kifejti, hogy mennyivel jobb annak a másiknak. És ezen aligha lehet segíteni, mert egyrészt „a szomszéd rétje mindig zöldebb", másrészt nem vagyunk egyformák, sem tudásban, sem gondolkodásmódban — vagy a témánál maradva: kódolási mentalitásunkban.

Való igaz, hogy a menyasszonyomat (programomat) nem szívesen cserélem le, mert megszoktam a gulyáslevesét és a viselkedését (mit minek alapján és hogyan csinál a program). És arra sincs garancia, hogy az új ara ugyanolyan finom túróscsuszát főz, vagy nem káromkodik még többet (hibaüzenetek). Esetleg én sem vagyok elég rugalmas, és nem tudok stílust váltani.

A programozónak attól függően kell a megfelelő szoftvereszközöket alkalmaznia, hogy milyen feladaton dolgozik. És az nagyon sokféle lehet! Hogy az egyes nyelvekben mennyire más programozási logikát kell követni, arról könnyen meggyőződhetünk (még programozói szaktudás nélkül is!), ha összehasonlítjuk ugyanannak a viszonylag egyszerű és rövid feladatnak több nyelven történő megoldását. A szakirodalomból összevadásztuk a Fahrenheit fokban megadott hőmérsékleti adatnak Celsius fokra történő átszámítására különböző nyelveken írt programokat és lemezmellékletünkön közreadjuk azokat. Látható belőlük, hogy azokon a nyelveken, amelyek szerkezetében fejrész és deklarációs rész is van, mennyivel hosszabb a program, Basic-ben pedig az egész csak két sor. Ebből azonban még nem lehet messzemenő következtetéseket levonni az egyes nyelvek alkalmasságára, hatékonyságára, kódolási munkaigényességére stb.

A programozónak mindig idejében szólnia kell(ene), ha a feladat szervezési paraméterei meghaladják vagy nehézkessé teszik a megoldást. Ezenkívül el kellene fogadnia azt is, hogy minden programrendszer általános módszerek és gyakorlati megoldások többrétű alkalmazását teszi lehetővé, a fejlesztő csoport ismereteinek függvényében. Egy-egy gyengécske megoldás láttán különösen a fiatalabb programozók szeretnek hetvenkedni olyasmivel, hogy „én ennél fél óra alatt jobbat írok". Amikor pedig tényleg meg kell azt csinálni, kiderül, hogy egy hét vagy egy hónap is kevés ahhoz, hogy ne csak legyen egy program, hanem hibátlan is legyen.

A „soknyelvűség" a programozás világában is egyre inkább alapkövetelmény. Egyes programrendszerek eleve különböző programnyelveken írt, de azonos adatbázist kezelő programokból állnak. És aligha nevezheti magát valaki igazi programozónak, ha nem tud a feladathoz alkalmazkodni.

Törley Dezső

6 hónap alatt ANGOLUL

A LONDON STÚDIÓ angol nyelvű Távoktató Programjának hat tankönyve és hat hangkazettája segítségével.

A London Stúdió angol nyelvű Távoktató Programja egy év alatt nagy népszerűségre tett szert. Ezzel a módszerrel egyre többen és egyre szívesebben tanulnak. Mi sem bizonyítja ezt jobban, mint a hozzánk érkezett több ezer levél. Többek között írtak nekünk váltott műszakban dolgozók, vidéken élő fiatalok, kismamák, elfoglalt üzletemberek, sorkatonák, mozgáskorlátozottak, nyugdíjasok, sportolók, ... tehát olyanok, akik időbeli, térbeli vagy anyagi korlátok miatt nem járhatnak rendszeresen nyelvtanfolyamra. Mivel a levelek döntő többsége azonos kérdéskört érint, arra gondoltunk, hogy a leggyakrabban feltett kérdésekre, ily módon, nyilvánosan válaszolunk:

• Napi egy órás tanulással egy könyvet egy hónap alatt lehet elsajátítani. • Fél év alatt, rendszeres napi tanulással a készség szintjéig begyakorolható a hat könyv tananyaga. • A tananyag több mint kétezer lexikai egysége a kifejezésekkel és az alapvető nyelvtani szerkezetekkel együtt adja alapját a szóbeli és az írásbeli kommunikációnak. • Egyedülálló abban, hogy a magyar gondolkodásmódon keresztül közelíti és magyarázza az angol nyelvhasználat szabályait és sajátosságait. • A tananyag egymásra épülő, részletes magyarázatokkal, megoldási kulcsokkal rendelkezik és a gyakorlati életben használatos szókincset gyakoroltatja. • A tankönyvek egyben munkafüzetek is.
• A tökéletes kiejtés, beszédértés és beszédkészség elsajátítását a hangkazetták speciális gyakorlatai segítik. • A hanganyag angol anyanyelvű előadók tolmácsolásában hallható, magyar nyelvű magyarázatokkal. • Ez a program minden korosztály számára eredményes.
• Közvetlen tanári segítség nélkül is eredményesen elsajátíthatja a tananyagot.
• Távoktatásunkban nincs teljesen magára hagyva, hisz tanulás közben folyamatos munkakapcsolatot tarthat fenn a London Stúdió szaktanáraival. • Egy-egy tankönyv elsajátítása után részt vehet hétvégi 12 órás kommunikációs tréningjeinken, ahol begyakorolhatja, illetve jobban megértheti az otthon tanultakat. A tréning vizsgával zárul. • Tréning nélkül is jelentkezhet vizsgára, ahol lemérheti nyelvtudását, s erről diplomát is kap. • Jó érzés újra és újra leírni, hogy mintegy 300 olimpikonunk is ebből a tananyagból tanult angolul.
• Másból tanulóknak és tanfolyamra járóknak is hatékony kiegészítés! • Vállalkozóknak, munkáltatóknak költségként elszámolható.



Nincs mentség a további halogatásra, mert ez a módszer bármilyen élethelyzetben és időbeosztás esetén is hatékonyan alkalmazható! E tankönyvek otthonába viszik az angol nyelviskolát.

AZ ANGOLTANÁR HÁZHOZ JÖN.

JUTALOMSORSOLÁS!
Amin az 1993. július 31-ig beérkező megrendelőszelvények vesznek részt.
A három fődíj:
egy-egy két hetes tanulmányi ösztöndíj Anglia egyik legjobb nyelviskolájában.

NYEREMÉNY- ÉS MEGRENDELŐSZELVÉNY
IGEN, szeretném megnyerni az angliai ösztöndíjat.
Londonban ❑ Oxfordban ❑ Cambridge-ben ❑ szeretnék tanulni.
IGEN, megrendelem a **LONDON STÚDIÓ** teljes távoktató tananyagát (6 tankönyv és 6 hangkazetta) ....példányban, **4.200,- Ft-os egységáron, utánvéttel.** A vételárat, a posta- és csomagolási költséget átvételkor fizetem.
Kérjük, hogy olvashatóan, nyomtatott nagybetűkkel töltse ki!
Megrendelését az alábbi címre küldje:
Euroklub Kft., 1114 Budapest, Fadrusz u. 12., 1425 Postafik 802.
Szállítási határidő: 6 hét • Kód: Alaplap

Név: ______________________
Cím: ☐☐☐☐ ____________ város
____________ utca ________ házszám
____
aláírás

1 2 3 4 5 6

INFORMÁCIÓKÉRÉS: A0564 ▶

Folyóiratok angol nyelven

Untangling public key cryptography (Új módszerek az adatátvitel titkosításában) Dr. Dobb's Journal (US) (1992/5)

LUC Public-Key Encryption (Nyilvános kódú titkosítás) Dr.Dobb's Journal (US) (1993/1)

Information encoding with two-dimensional bar codes (Információkódolás kétdimenziós vonalkódokkal) Computer — IEEE (US) (1992/6)

Bar codes break out (A vonalkód-technológia forradalmi újdonságai és alkalmazási lehetőségei a kereskedelmi, raktárgazdálkodási, adatgyűjtési és termelési feladatok megoldásában) Datamation (GB) (1992/7)

A shared resource access manager, part 1 (Az osztott hálózati erőforrások vezérlésének és a hozzáférési lehetőségek koordinálásának problémái többfelhasználójú rendszerekben — 1. rész) Byte (US) (1992/8)

Selection from comp.compilers (Válogatás a compiler-compilerek problémáiból) SIGPLAN notices (US) (1992/1)

Plugging into the Internet (Az Internet-hálózathoz való kapcsolódás módjai: telefonvonali, terminálemulációs, illetve TCP/IP hálózati adatátviteli kapcsolat) Byte (US) (1992/10)

Data access solution now has IBM road map (Egységes hozzáférés különböző adatbázisokhoz: az IBM Information Warehouse adatbázis-termékének architektúrája és a transzparens adathozzáférés kérdései) Software Magazine (GB) (1992/2)

Database mix poses a sharing challenge (Osztott adatbázisok alkalmazása) Software Magazine (GB) (1992/3)

GPF 2.0: first 32-bit GUI development tool for OS/2 (GPF, Version 2.0: OS/2 2.0-ás verzióhoz készült első 32 bites grafikus interfész fejlesztőeszköz) PC Magazine (US) (1992/17)

The evolution a C++ programmer (Áttérés a C-nyelvű programozásról a C++ nyelvű objektumorientált programozásra) Computer Languages (GB) (1992/8)

Windows development tools: Part 2 — Product wrap-up (Windows szoftverfejlesztő eszközök: 2. rész) Computer Languages (GB) (1991/3)

ASCII goes global (Törekvések az egységes, nemzetközileg elfogadott betűkódszabvány megteremtésére) Byte (US) (1991/7)

Graphic display of larger sentence dependency structures (A Texnet szövegszerkezet-megjelenítő prototípus-rendszer bemutatása: az automatikus dokumentálás lehetőségeinek kutatása) Journal of the American Society of Information Science — IASIS (US) (1991/ 5)

Data structures (Part 2) — Doctor C's pointers (Adatstruktúrák C nyelven (2. rész) többdimenziós tömbök, karaktersorozat-elemű tömb, osztott és minősített hozzáférés) The C Users Journal (CA) (1991/5)

Single-chip decoder PLDs give users freedom of choice (A Cypress Semiconductor cég 50 MHz-es BiCMOS programozható logikai dekódoló áramkörei) (1991/8)

The JPEG still picture compression standard (A JPEG digitális képi információtömörítő szabvány) Communications of the ACM (US) (1991/4)

MPEG: a video compression standard for multimedia applications (Az MPEG: video(kép) tömörítő szabvány multimédia-alkalmazások számára) Communications of the ACM (US) (1991/4)

Overview of the PX 64 kbit/s video coding standard (A PX 64 kbit/s video kódolási szabvány áttekintése) Communications of the ACM (US) (1991/4)

CD-I full-motion video encoding on a parallel computer (A CD-I — Compact Disc-Interactive — egyik lehetséges alkalmazási területe a teljes körű mozgó videotárolás) Communications of the ACM (US) (1991/4)

Automated generation of code using backtracking parsers for attribute grammars (Attribútum-nyelvtanok automatizált kódgenerálása az elemzés visszirányú nyomkövetésének alkalmazásával) Sigplan Notices (US) (1991/2)

Searching in discrete universes (Keresési algoritmusok a nyelvtani elemzés, kompilálás és kódgenerálás során) Sigplan Notices (US) (1991/3)

A dynamic accessibility protocol for replicated databases (Ismétlődő adatokat tartalmazó adatbázis dinamikus hozzáférési szabványa) Data and Knowledge Engineering (NL) (1991/4)

Encoding Japanese characters (Japán karakterek kódolása) The C Users Journal (CA) (1991/7)

The Art of reverse engineering (A szoftver 'reengineering' művészete) Computer Languages (GB) (1991/6)

Decoding ASN.1 transfer syntax (Az ASN.1 adatátviteli nyelvtan kódolása) The C Users Journal (CA) (1991/9)

Application generators — Product wrap-up (Négy MS-DOS alatt futó C nyelvű alkalmazásgenerátor bemutatása) Computer Languages (GB) (1991/9)

A bandwidth reducing token ring (Csökkentett sávszélességű token-ring hálózat) Computer Networks and ISDN System (NL) (1991/4)

Security in open networks and distributed systems (Biztonsági kérdések nyitott hálózatokban és osztott rendszerekben) Computer Networks and ISDN System (NL) (1991/5)

A model of authorization for next-generation database systems (Hozzáférési jogosultság modellezése a jövő adatbázisrendszereiben) ACM Transactions on Database Systems (US) (1991/1)

Data compression using word encoding with Huffmann Code (Nagyméretű könyvtári adatbázisok rekordjainak tömörítése Huffman-kódolással — minimum-redundancy codes) Journal of the American Society of Information Science — IASIS (US) (1991/9)

Why the OMG Object Request Broker should mean good news for object databases (AZ OMG hálózat ORB objektumkezelő szoftver jelentősége az objektumorientált adatbázis-kezelésben) Journal of Object-Oriented Programming (US) (1991/4)

Parallel access to an English dictionary (Öt transzputerre elosztott angol nyelvű számítógépes szótár párhuzamos elérése) Microprocessors and Microsystems (GB) (1991/6)

Find the source — product wrap-up (Öt disassembler ismertetése) Computer languages (GB) (1991/11)

Chinese Lotto as an exhaustive code-breaking machine (Keresés, kódmegfejtés osztott számítógéprendszerben) Computer — IEEE (US) (1991/11)

Register usage strategies (A C nyelvű fordítók regiszterhasználata) The C Users Journal (CA) (1991/11)

Prolog — relational database interfaces (A Prolog nyelv interfészei relációs adatbázisok eléréséhez) (1991/10)

Surface coding based on Morse Theory (Felületek kódolása a Morse-elmélet alapján) Computer Graphics and Application Magazine (US) (1991/5)

Robust string matching (Megközelítőleg azonos karaktersorozatok kiválogatása szöveges adatokból) Computer Languages (GB) (1991/12)

A high-performance software implementation of the Data Encryption Algorithm (DEA) (ANSI-szabványú adattitkosítási algoritmus szoftveres megvalósítása) Journal of Microcomputer Applications (GB) (1991/4)

Folyóiratok német nyelven

Netz-Werkzeugkasten (Szoftverfejlesztő környezet neurális hálózatok és fuzzy logika számára) Elektronik (DE) (1992/2)

Der DinBus als Feldbussystem für Zeiterfassung, Zugangskontrolle oder Produktionssteuerung (DinBus DIN 66 384 szerinti sínrendszer vonal- és mágneskódos rendszerekhez) Elektronik (DE) (1992/7)

BTX unter Windows: Vier Decoder (Windows-zal használható képernyőszöveg dekóderprogramok összehasonlítása) Chip (DE) (1992/3)

Risiko Computer: Schutz vor Saboteuren und technischen Defekten (Az adatvesztési lehetőségek csoportosítása és elhárításának módszerei) Chip (DE) (1992/4)

Hardwareschutz: Harte Nüsse für Langfinger (Hardvermegoldású szoftvervédő modulok bemutatása) Chip (DE) (1992/4)

Kostengünstig über Zweidrahtleitungen (ISDN-állomások hagyományos kétvezetékes alkalmazások céljára) Elektronik (DE) (1992/11)

Elektronik Beschleunigt Postzustellung (Postai küldemények távkódolása) Elektronik (DE) (1992/16)

Videotextdecoder für alle Amigas, Folge 1 (Videotext-dekódoló az összes Amigához, 1. rész) Amiga (DE) (1992/4)

Videotext-Decoder für alle Amiga, Folge 2 (Videotext-dekódoló az összes Amigához, 2.rész) Amiga (DE) (1992/5)

Multitalent für 32-Bit-Code: Watcom C/386 Version 9.01 (Watcom C/386 Version 9.01: 32 bites compiler MS-DOS, Windows és OS/2 operációs rendszerek számára) MC — Die Mikrocomputer-Zeitschrift (DE) (1992/9)

Datenkompression: gut gepackt ist halb gewonnen (A tömörítőprogramok működési elvei és programozásuk) Computer Persönlich (DE) (1991/16)

Hardware-Decodierung beschleunigt VXI-Systeme (A soros VXI-sínrendszer működésének meggyorsítása hardver dekóderrel) Elektronik (DE) (1991/16)

Kompakte Codierverfahren (Kompakt kódolási eljárások) Elektronik (DE) (1991/20)

Plus für Bildschirmtext (BTX/2 Plus: teletexdekódoló szoftver az NSZK-beli Bildschirmtext szolgáltatásához) BIT — Büro- und Informationstechnik (DE) (1991/7/8)

Bild-kompressor: Codierung nach CCITT (CCITT-szabvány szerinti képtömörítés) MC — Die Mikrocomputer-Zeitschrift (DE) (1991/10)

Datenkopression: Die Bits rücken zusammen (Az adattömörítés alkalmazási területei és algoritmusai) Computer Persönlich (DE) (1991/23)

Betriebssystem-Erweiterungen mit Passwortschutz: PC-Mentor und Cockpit (Jelszavas adatvédelmet biztosító operációs rendszer kiegészítő programjainak bemutatása) MC — Die Mikrocomputer-Zeitschrift (DE) (1991/12)

Könyvek — magyarul

Halassy Béla — Zentai Tamás: Döntési táblázatok. Budapest, 1973. SZÁMOK, 169 oldal.

Németh József: Adatvédelem a számítógépes és hírközlő rendszerekben. Budapest, 1984. Számítástechnika-alkalmazási Vállalat, 157 oldal.

Nemetz Tibor — Vajda István: Algoritmikus adatvédelem. Budapest, 1991. Akadémiai Kiadó, 240 oldal.

Pajzs Júlia: Számítógép és lexikográfia. Budapest, 1990. Az MTA Nyelvtudományi Intézete, 83 oldal.

Ian Stewart: A matematika problémái. Budapest, 1991. Akadémiai Kiadó, 280 oldal.

Titokvédelem számítástechnikai környezetben. Budapest, 1990. SZÁMALK, 63 oldal.

Vargha Balázs: Játékkoktél. Budapest, 1967. Minerva, 291 oldal.



Sun Race

a Nap kedvence

Mit tud ma egy notebook?

Szorozza meg kettővel!

Mennyibe kerül ma egy notebook?

Ossza el kettővel!

A Sun Race notebook-ot az igények szülték.
Az Ön igényei.



9028 GYŐR FEHÉRVÁRI U.80.
TEL.: (96) 17-722 FAX: (96) 17-943

1139 BUDAPEST LOMB U.37.
TEL.: 120-2602 FAX: 120-2672



-színes 10"LCD,és egyidejű képernyő kezelés-
-SCSI port-szivargyújtóhoz kapcsolható-
-jelszavas hozzáférés-energiaszint jelző-
-turbó akkutöltő-numerikus tasztatúra-
-beépített Logitech TrackBall-
-hordtáska,dokumentáció-cserélhető winchester-

486/SLC notebook,120 MB HDD,4MB RAM,257.650 Ft-tól

INFORMÁCIÓKÉRÉS: A0479 ▼

SZÁMÍTÁSTECHNIKA KULCSRAKÉSZEN!

SZÁMÍTÓGÉPEK, NYOMTATÓK, MODEMEK SZÉLES VÁLASZTÉKA!

- AT 286-os, 386-os, 386SX, 486-os számítógépek minden kiépítésben (3 ÉV GARANCIÁVAL!)
- Laptop, notebook gépek
- EPSON, Hewlett–Packard nyomtatók, perifériák
- DISCOVERY és US ROBOTICS MODEMEK és távadatátviteli rendszerek
- APC szünetmentes tápegységek
- SZOFTVEREK és SHAREWARE-ek teljes választéka
- NOVELL HÁLÓZATI SZOFTVEREK, hálózatépítés

Kérjük, telefonáljon vagy írjon, és mi örömmel adunk felvilágosítást, küldünk részletes árjegyzéket!

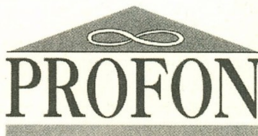
QWERTY

High Tech Kft.
Bemutatóterem: 1114 Budapest, Bartók Béla út 9.
Telefon: 18-68-858, 18-52-687, 16-63-098
Telefax: 18-52-687
BBS: 11-87-950 BUDAPEST BBS

NE FELEDJE: Nevünk ott található az ÖN számítógépének billentyűzetén is!

INFORMÁCIÓKÉRÉS: A0529 ▼

KÁBELHÁLÓZATOK



HELYI KÁBELHÁLÓZATOK
tervezése és kivitelezése

ADATHÁLÓZATOK
- IBM Cabling System
- ETHERNET
- UTP
- Twinaxiális
- Koaxiális
- Egyéb

ERŐSÁRAMÚ HÁLÓZATOK
- Számítástechnikai rendszerekhez

HÍRKÖZLŐ HÁLÓZATOK
- Alközponti hálózatok
- Modemes hálózatok

RACKSZEKRÉNYEK
RACKSZERELVÉNYEK
ÖSSZEKÖTŐ KÁBELEK

1141 Budapest, Egressy út 113/E
Telefon/Telefax: 252-0663

INFORMÁCIÓKÉRÉS: A0141 ▼

Látni és láttatni IV.

Hasonlóság és különbözés szerint

Az előző részben a látás automatizálásának képfeldolgozási, -osztályozási és -felismerési technikái közül az „első etappal", az előfeldolgozás képjavító módszereivel ismerkedtünk. Ha már elégedettek vagyunk a kép minőségével, a feldolgozás további lépése, hogy elkülönítjük az értékes információt hordozó képpontokat a háttértől — ezt a műveletet nevezzük szegmentálásnak. Amikor egy képpontot értékesnek minősítünk, tulajdonképpen már osztályozzuk is a képet.

A szegmentáló eljárásokat most két okból is érdemes külön tárgyalni:

— mert az osztályozáshoz a színkódokat vesszük alapul;

— és mert eredményül (geometriai értelemben) összefüggő ponthalmazokat, vagyis objektumkat keresünk (az „igazi" osztályozást majd az objektumok sajátságai alapján végezzük).

A szegmentálással foltokhoz, illetve élekhez jutunk. A foltokat olyan képpontok alkotják, amelyek vizsgált tulajdonságaik tekintetében hasonlók a környezetükhöz; az éleket (a foltok határvonalait) viszont épp ellenkezőleg úgy találjuk meg, hogy olyan képpontokat keresünk, amelyek valamiben különböznek a szomszédaiktól.

Itt most megint el kell egy kicsit gondolkoznunk. A hasonlóság, illetve különbözőség ugyanis nehezen mérhető.

Miként tudjuk a számítógépet megtanítani rá, mikor tekintsen két képpontot elég hasonlónak vagy eléggé különbözőnek ahhoz, hogy egy folthoz vagy egy élhez tartozónak minősítse — szaknyelven szólva: egy osztályba sorolja — őket?

„Sajátságos" egyedek

Nyilvánvalóan kell találnunk valamilyen számszerűsíthető jellemzőket — ezeket nevezzük sajátságoknak. (Ne feledjük, hogy a digitális kép egy számmátrix; ezért a kép bármilyen jellemzőjét csak ezekből a számokból lehet és kell kiszámítani, matematikai műveletekkel.)

De ez még nem elég. Mondjuk, hogy valamilyen szabály alapján meghatároztuk két szomszédos képpontnak valamilyen sajátságát. Hogyan döntjük el ezek alapján, hogy eléggé hasonlóak-e, vagyis egy osztályba tartoznak-e? Szükségünk van egy képletre is, amellyel kiszámíthatjuk két sajátság „távolságát"; ezt nevezzük távolságfüggvénynek (idegen szóval: metrikának).

Végül meg kell adnunk még egy küszöbszámot is. Ezek után megfogalmazhatjuk a gép számára a döntési szabályt: két képpontot akkor sorolj egy osztályba, ha a megfelelő sajátságuk távolsága kisebb (illetve nagyobb) a megadott küszöbnél.

Fontos: a sajátságok „távolságának" semmi köze a képpontok geometriai távolságához! Két szomszédos képpont sajátságai lehetnek olyan „messze" egymástól, hogy különböző osztályokba tartozzanak — és fordítva: a kép két legtávolabbi sarkában levő képpontok sajátságaik „közelsége" révén tartozhatnak egy osztályba.

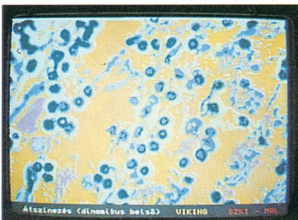
Csak egy kis matematika

Nagyon is gyakorlati példaként nézzük meg, hogyan lehet egy valódi színes képet megjeleníteni egy VGA képernyőn. Az előbbiben — 8 bites színkódok esetén — közel 17 millió féle (2 — hatvány:24=16 777 246), az utóbbin 256 szín lehetséges. A probléma megoldásához szerkesszünk egy olyan 3 dimenziós koordináta-rendszert, amelynek tengelyeire — a matematikában megszokott hosszúságok helyett — színkódokat mérünk fel (1. ábra). Mivel most minden képponthoz 3 sajátságot veszünk figyelembe, nevezetesen a 3 színkódját, az ily módon létrehozott színtérben mindegyiknek egy sajátságpont felel meg, amelyik ez esetben éppen a színét határozza meg.

Az ábrán a (160, 96, 32) „koordinátájú" P pont egy sötét narancssárga színű képpont sajátságpontja. Milyen messze van ez például az M (255, 255, 0) sárga színétől? A vastag vonallal jelölt euklidészi távolságot a Pitagorasz-tétel segítségével lehet kiszámítani, míg a vastag szaggatott vonallal rajzolt négyes-távolság egyszerűen a



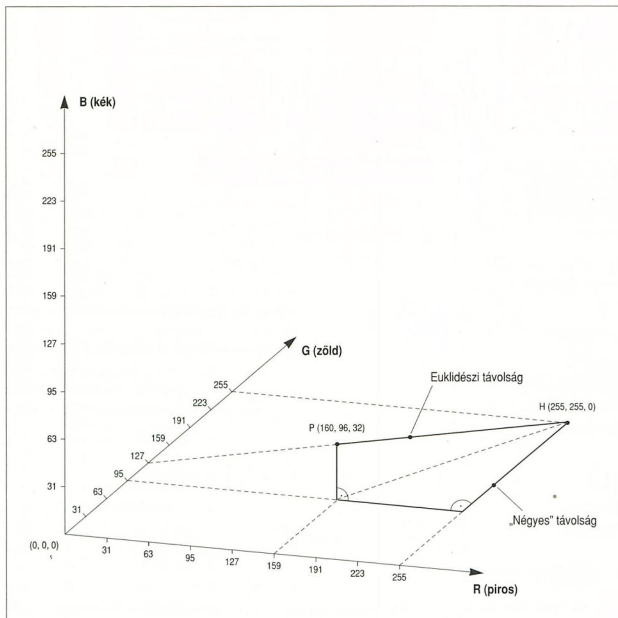
1.kép. Javított mikroszkópi felvétel álszínesen



2.kép. 5 színtre vágott kép



3.kép. 2 színtre vágott kép



1. ábra. Távolságmérés a színtérben

koordinátakülönbségek abszolút értékének összege. Természetesen sok más távolságfüggvény is lehetséges.

Akit nem riasztanak a képletek, azok számára felírjuk a két metrikát is:

$$de(M,P) = \sqrt{(rM-rP)^2 + (gM-gP)^2 + (bM-bP)^2};$$

illetve

$$d4(M,P) = |rM-rP| + |gM-gP| + |bM-bP|.$$

Itt de, ill. d4 az euklidészi, illetve a négyes-távolság; r,g,b pedig a pontok színkoordinátáit jelölik, a megfelelő indexszel.

Melyik az igazi? Ez az ártatlannak és természetesnek látszó kérdés nehéz elvi-matematikai és sokszor filozófiai problémákat takar. A metrika és a döntéshez szükséges küszöbök helyes — a megoldandó problémához legjobban illeszkedő — megválasztása ugyanis döntően befolyásolja az osztályozás „jóságát". Rossz választás teljesen hamis, a valóságtól merőben eltérő eredményhez vezethet.

„Jószomszédi" viszony

Térjünk most vissza az átszínezési problémához. A „legközelebbi szomszéd" módszer alapján a következőképpen kellene eljárnunk:

— először is kijelöljük a színtérben a megengedett 256 színnek megfelelő sajátságpontot;

— ezután kiszámítjuk minden egyes képpont sajátságpontjának távolságát az összes megengedett színétől, és mindegyiket ahhoz a színosztályhoz rendeljük, amelyikhez a legközelebb van;

— végül minden képpontot átszínezünk az osztályának megfelelő színre.

A rengeteg számolást leegyszerűsíthetjük, ha inkább a „doboz" módszert alkalmazzuk. Állapodjunk meg abban, hogy az eredő VGA színkódban 3-3 bitet szánunk a piros és a zöld, 2 bitet pedig a kék színösszetevő ábrázolására. Ennek megfelelően osszuk fel a piros és a zöld tengelyt 7 küszöbbel 8-8, a kéket 3 küszöbbel 4 egyenlő részre, majd az osztáspontokon át fektetett, a koordinátasíkokkal párhuzamos síkokkal a színteret 256 „színdoboz"-ra (2. ábra). Ezek után minden képpontot olyan színűre színezzünk, amilyen „színű" dobozba esik a sajátságpontja.

Anélkül, hogy belevesznénk az igazságkeresés filozófiai útvesztőibe, ebben a konkrét esetben nyugodtan elfogadhatjuk igaznak azt az eredményt, amely nem ellenkezik az átlagos emberi látórendszer tapasztalatával. Ha a VGA-képen az eget kéknek, a füvet zöldnek és a piros rózsát pirosnak látjuk, akkor jól osztályoztunk.

Dobozolás helyett vágás

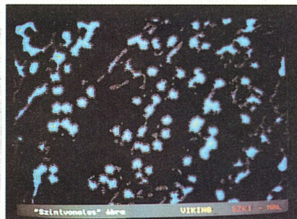
Meg kell mondani őszintén, hogy a doboz módszer nem az igazi, ugyanis burkoltan azon a feltevésen alapul, hogy a színtér homogén és anizotróp. (Magyarul: a sajátságpontok minden irányban egyenletes sűrűséggel helyezkednek el.)

Ez távolról sincs így.

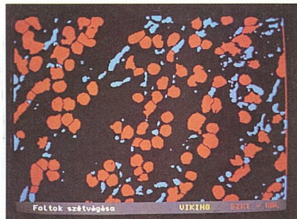
Világosságérzékelésünk ugyanis logaritmikus, színérzékelésünk pedig még bonyolultabb: különböző irányokban a színezettől függően is változik. Elfogadható eredményt csak úgy lehet kapni, ha olyan metrikát agyalunk ki, amelyben a színkoordináták megfelelően súlyozva vannak, a távolságképlet valamilyen exponenciális függvény, és



4.kép. Képvágási hibák eltávolítása



5.kép. „Szintvonalkép" az objektumok szétvágásához



6.kép. Megtalált objektumok

INFORMÁCIÓKÉRÉS: A0534 ▼





Microsoft®

BORLAND

COREL



SPC

LOGITECH

MICROGRAFX®

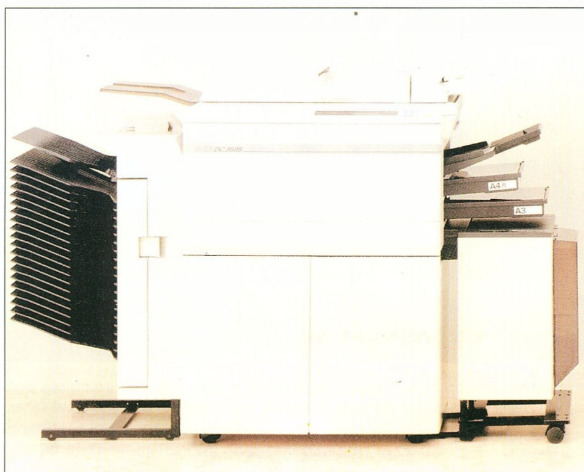


CompMark

Számítástechnikai és Kereskedelmi Kft.
1138 Budapest, Párkány u. 20.
Telefon/Telefax: 173-1272, 173-1358

INFORMÁCIÓKÉRÉS: A0545 ▼

COPY–SYSTEM KERESKEDELMI ÉS SZOLGÁLTATÓ KFT.



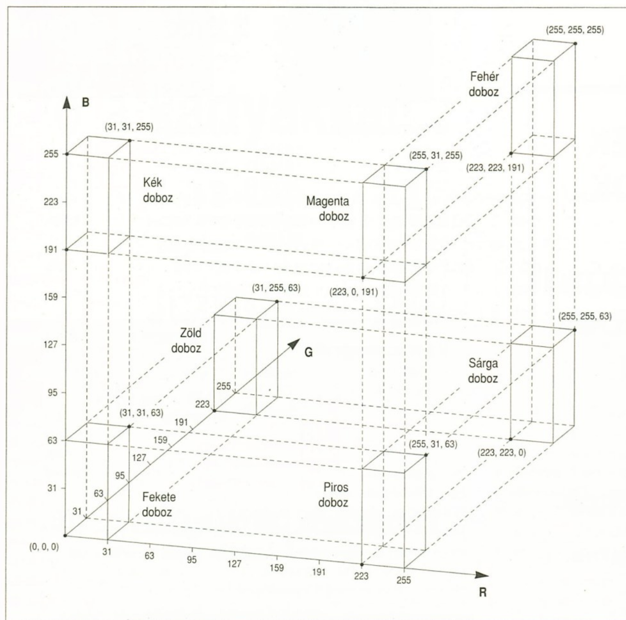
mita

MÁRKASZERVIZ

MITA, REX-ROTARY,
GESTETNER, U–TAX
MÁSOLÓGÉPEK JAVÍTÁSA
KELLÉKEK, ALKATRÉSZEK
ÁRUSÍTÁSA
VISZONTELADÓKNAK IS

1067 Budapest, Eötvös utca 47. • Telefon: 111-1676 • Telefax: 111-4836

INFORMÁCIÓKÉRÉS: A0206 ▼



2. ábra. A színtér felosztása 256 „színdobozra"-ra

a küszöbök sem egyenlő közűek, hanem színtől és iránytól függően változnak. Mindez azonban már egy másik történethez tartozik.

Szerencsére a gyakorlatban sokszor lehet kielégítő szegmentálási eredményeket kapni egy egyszerűbb módszerrel is, amelyet (kép)vágásnak nevezünk.

Ez nem más, mint a kép küszöb(ök) szerinti átszínezése. Az eljáráshoz eggyel kevesebb küszöböt kell megadni, mint ahány osztályt ki akarunk alakítani.

A „vágás" úgy történik, hogy az i-edik osztályba soroljuk azokat a képpontokat, amelyek színkódja nem nagyobb az i-edik küszöbnél; az utolsó osztályba pedig az összes még be nem soroltat. (Vizuális kiértékelés céljából az egy osztályba sorolt pontokat azonos színűre szokás átszínezni, ugyanakkor ez a gépi feldolgozáshoz nem szükséges.)

A módszer akkor vezet jó eredményre, ha a kép hisztogramjában jól elkülöníthető csúcsok vannak, viszonylag mély völgypontokkal elválasztva. Küszöbül a völgyek mélypontjainak megfelelő színkódo(ka)t célszerű választani.

A folyamat és a látvány

Az 1. képen a sorozat II. részében már mutatott mikroszkópi sejtfelvételnek egy zajszűréssel és háttér-kiegyenlítéssel javított változata látható. Egy automatikus vágóprogram (a hisztogram elemzése alapján) 4 küszöböt talált, és ennek megfelelően 5 szintre vágott — 2. kép.

A további feldolgozáshoz azonban ez túl részletes volt, ezért a sötétkék és a fekete, valamint a világoskék, a lila és a sárga osztály összevonásával a 2 szintre vágás is elég — 3. kép.

Az értékes információt a sejteknek megfelelő fekete színű objektumok hordozzák, a képvágás globális jellege miatt azonban egyikük-másikuk belsejében lokális hibák (háttérfoltok) maradtak. A következő lépésben ezeket kerestük meg, és hozzászíneztük a sejtekhez — 4. kép; itt még a megtalált hibák kék színűek.

Ezután egy ügyes programmal megkíséreltük szétválasztani az érintkező objektumokat.

Az eljárás feltételezi, hogy a sejtek konvex alakúak. Ezért először kiszámítja az objektumok belső pontjainak távolságát a határpontoktól, majd az azonos távolságra levő pontokat azonos — befelé haladva egyre világosabb — színűre színezi át.

Ily módon egy szintvonalas térképhez hasonló kép keletkezik — 5. kép. Ha egy objektum belsejében két vagy több világos „csúcs" alakul ki, a program a köztük levő „nyeregvonal" mentén vágóvonalat generál.

A foltok létrehozására irányuló szegmentálás végeredményét szemlélteti a 6. kép, amelyen az értékes objektumok piros színűek, és köztük jól láthatók a vágóvonalak. A sejtnek nem minősíthető, szabálytalan alakzatokat a program kékre színezte, ezek a további feldolgozásban már háttérnek minősültek.

Fontos megjegyezni, hogy a szegmentálás eredményét csak a látványképen színezzük át (rendszerint a kimeneti átszínezőtáblák megfelelő feltöltésével), a jobb megjelenítés kedvéért. A gép azonban pontosan „tudja" az egyes objektumokhoz tartozó képpontok eredeti színkódját, s ezeket a továbbiakban fel is használja.

Álló Géza

CONET

Számítógép és Hálózat
Rendszerfejlesztő Kft.
1142 Budapest, Kassai u. 67.
Telefon: 163-6046, 163-6047
Telefax: 251-0721

SZÁMÍTÓGÉPEK és HÁLÓZATOK területén komplett megoldások a világ élvonalbeli termékeiből.

IBM

PS/1, PS/2 gépek
Token Ring, Cabling System,
Twinax hálózatok,
Novell, Ethernet hálózatok.

DEC

VAX, MicroVAX gépek,
munkaállomások,
DECnet hálózatok,
Ethernet, FDDI hálózatok.

HÁLÓZATÉPÍTÉS

Kábelezés,
IBM LAN eszközök,
DEC LAN eszközök,
XYPLEX LAN eszközök,
RAD, RND, LANNET
kommunikációs berendezések.

NAGY TERÜLETŰ HÁLÓZATOK

Bridge-ek, repeaterek,
multiplexerek,
távoli összeköttetések,
Backbone rendszerű hálózati
csomópontok,
SNMP hálózatmenedzserek.

INFORMÁCIÓKÉRÉS: A0506 ▲



MICROLINE Kft.

1081 Budapest,
Szilágyi u. 1. fszt.
Telefon: 113-4442
Telefon/Telefax: 114-6813

Winchesterek:

40 MB-os	14935 Ft
80 MB-os	19360 Ft
120 MB-os	24000 Ft
170 MB-os	27710 Ft
240 MB-os	34722 Ft
540 MB-os	95330 Ft

Floppyk:

1,2 MB-os	5260 Ft
1,44 MB-os	4100 Ft
CD–ROM MITSUMI	22300 Ft

Koprocesszorok:

IIT 2C87–20	4560 Ft
IIT 3C87SX–33	7300 Ft
IIT 3C87DX–40	8160 Ft

CPU-k:

486DX2–50	46950 Ft
486DX2–66	62270 Ft

Alaplapok:

386SX–33	9220 Ft
386DX–40	16500 Ft
486DX2–50	59650 Ft
486DX2–66	74970 Ft
3486 VESA BUS	13880 Ft

Kártyák:

IDE 2S/P	1240 Ft
VGA REALTEK 3105	1840 Ft
VGA OAK 087, 0 kB WINDOWS ACC	3500 Ft
IDE CACHE CONTROLLER, 0 kB	16160 Ft

RAM memóriák:

44256–80	400 Ft
SIMM, 256 kB	935 Ft
SIMM, 1 MB	3000 Ft
SIMM, 4 MB	13200 Ft

AZ ÁRAK NEM TARTALMAZZÁK AZ ÁFÁT.
Az árak változtatásának jogát fenntartjuk.

INFORMÁCIÓKÉRÉS: A0518 ▲

star lézernyomtatók,
Canon lézernyomtatók, fénymásolók

ÁRUSÍTÁSA

MAGYARORSZÁGON A LEGOLCSÓBBAN

ÚJ FESTÉKKAZETTÁK

HP és Canon típus:
8000–12500 forint + áfa

KAZETTÁK FELÚJÍTÁSA

4500–6500 forint + áfa

USA technológiával – kék és barna színben is

KEDVEZMÉNYEK: mennyiségi, törzsvásárlói
INGYENES kiszállítás (Budapest területén)
Árusítás, újratöltés utánvéttel is.

LÉZERNYOMTATÓK KARBANTARTÁSA

Készpénzre beváltható felújításkor.



CompuDrug Standard Kft.

Cím: Budapest X., Népliget, Planetárium
Telefon: 133-1576

INFORMÁCIÓKÉRÉS: A0528 ▲

Amiről a PC zenél I.

Elő a kártyákkal!

Nem is olyan régen még jogos volt a C64- és Amiga-tulajdonosok büszkesége, ha gépüket „zenei képességek” szempontjából hasonlították össze a PC-kkel. Miközben a PC-k teljesítménye minden más szempontból nagyobb volt, „zeneileg” sípoláson és brekegésen kívül csak hörögni tudtak. Ez pedig a hangképzés terén bizony édeskevés... Ma már azonban minden alaposan megváltozott. Mostani cikkünkben és a következő számban közlendő folytatásban bemutatjuk az új lehetőségeket.

Eredetileg a PC-k pár centiméter átmérőjű, maximum negyed wattos zsebrádió hangszóróján megszólaló egybites jelforrása csak arra volt elegendő, hogy egy művelet befejezésekor figyelmeztető jelzést adjon: ő készen van, újabb parancsra vár. Bizonyos speciális szoftverekkel a PC ugyan megtanult egy kicsit zenélni és beszélni, de egyrészt még mindig egy biten, másrészt a hang minősége maximum egy lerobbant tölcséres gramofonéra emlékeztetett. Érthető, hogy akik Commodore gépek mellől ültek át a PC-hez, gyakran elmerengtek azon, hogy korábban milyen jó, többszólamú kísérőzenéik voltak, most pedig jobb esetben is csak csönd van.

A PC-k nyitott architektúrája azonban kezdettől fogva magában hordozta a hangképzés lehetőségét. A fejlesztő cégek végül az utóbbi időkben egymással versengve dobták piacra egyre jobb képességű hangkártyáikat és szoftvereiket. Ezekkel felszerelve a PC hangképzése már felülmúlja az Atari és Macintosh gépcsaládokét, amelyek általában már alapkiépítésükben 3-4 darab 8 bites D/A konverterrel készültek. A zenei alkalmazások terén az Atari vitathatatlan fölénye alighanem a múlté. A MIDI-vel a PC karrierje meredeken felfelé ívelő szakaszához ért (MIDI = music instrument digital interface).

Az utóbbi évek „hangos” termékeinek teljes áttekintéséhez persze egy egész újság sem lenne elegendő. A piac kínálata igen széles skálán mozog, mind a hobbi, mind a professzionális kategóriában. Ezért — a teljesség igénye nélkül — kiválasztottunk néhány hardver- és szoftvereszközt, amelyek a magyar piacon a legnagyobb népszerűségnek örvendenek. Ezekhez a forrásinformációkat a Pixel Graphics Kft. és a Kerorg bocsátotta rendelkezésünkre, ők az alább ismertetendő termékek döntő többségének magyarországi disztribútorai. (Sajnos a Midisoft Kft. nem élt a bemutatkozás lehetőségével.)

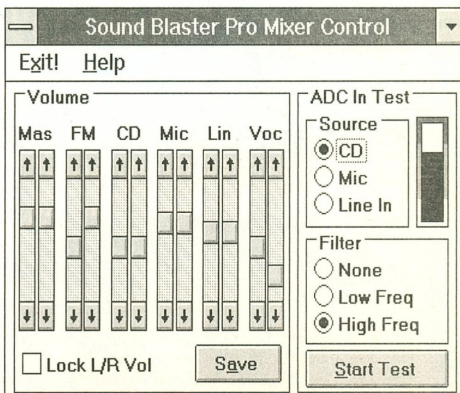
Amire figyelni kell

Ha valamelyik kártya vagy program megfelel igényeinknek, ne mulasszuk el pontosan tisztázni, hogy milyen feltételek között hajlandó azt a teljesítményt nyújtani. A kártyák installálása során nem nagyon ütközünk problémákba. Annál inkább oda kell figyelni, hogy a kiválasztott kártyához van-e saját szoftver és milyen, hogy melyik másik szoftverrel lehet még futtatni, működik-e a szoftver DOS alatt vagy csak Windows-környezetben, jó-e neki a 286-os alaplap 1 MB RAM-mal, vagy többet kíván stb.

A PC-kbe rakható hangkártyákon található egy MIDI-interfész vagy egy szintetizátor, vagy egy sampler (mintavételező) — esetleg mindhárom egyszerre. A hangkártyákat működtető szoftverek általában több funkciót látnak el. Ezek a következők: sequencer (szekvenciaszerkesztő), voice editor (hangszerkesztő), notator (kottagrafika-szerkesztő), sampler (külső hangforrást felvevő mintavételező). Az utóbbi forrás lehet akár mikrofon, CD, magnó vagy MIDI-hangszer is.

A legegyszerűbb hangkártyák csak egy MIDI-interfészt tartalmaznak. Ennek egyik őstípusa a Roland MPU-401 (MPU = midi processing unit) kártya. Gyakorlatilag minden kártya, amelyik ad magára, ezzel kompatibilis. Jelenleg a piacot ennek egy továbbfejlesztett változata, a Roland MPU-IPC-T uralja, de egyre nő a hasonló tudású Voyetra kártyák (V-22m, V-24sm, V-4000) népszerűsége is.

Az MPU-IPC-T egy 8 bites félhosszú kártyára épült, egy 25 pólusú Canon csatlakozóval kapcsolódik a külső szinkronáramköröket tartalmazó dobozhoz. Ezen egy MIDI In és két MIDI Out 5 pólusú DIN-csatlakozó, egy magnószinkron, egy magnókimenet és egy 3,5-es jack metronóm hangkimenet van. A Voyetra V-22 közel ugyanezt a szolgáltatást nyújtja, a V-24 két MIDI-portot is tartalmaz, így egyszerre 32 csatornát is tud kezelni, a V-4000 pedig négy MIDI-porton 64 csatornát.



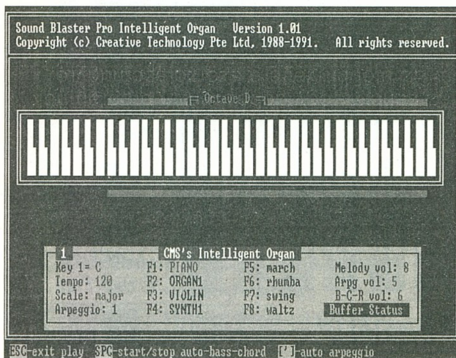
Az első szintetizátortól...

Az AdLib a PC-khez fejlesztett szintetizátorkártyák őstípusa. A kártya 11 csatornás, 2 operátoros FM (2 generátoros szinuszoszcillátoros hangszintetizátor) hangját egy 4 wattos mono belső erősítő segítségével szólaltatja meg. (A hang „szépsége", hűsége a valósághoz a generátorok számától függ.) A szoftverek többsége — mivel felülről kompatibilisek ezzel — együttműködik vele. A kártyát „ismerő" játékok hangjának megszólaltatására bevált, bár éles, száraz (túl tiszta szinuszos) hangja miatt nem mindenki kedveli.

A BA-2000 kompatibilis az AdLibbel, két üzemmódjában vagy 9 dallamhangot, vagy 6 dallamhang mellett 5 ütőhangszer hangját szólaltatja meg. Ez is 2 operátoros, egy időben 128 hangprogram választható ki rajta. A PC hangszórójára is közvetlenül ráköthető. A BA-3000 plusz szolgáltatása az, hogy egy 8 bites A/D bemenete is van. A digitalizált hangmintát megfelelő szoftverrel a winchesterre írja, így a felvett hang(ok) tartamát csak a háttértároló mérete korlátozza.

A Sound Blaster (Sound Buster — tajvani változat) 2.0 alaptulajdonságaiban megegyezik a BA-3000-rel. Van rajta viszont MIDI-interfész, amelyet 9 tűs csatlakozón keresztül csatlakoztathatunk egy MIDI IN és OUT vonalra. A bemenet mikrofon és line (pl. magnó, mikrofon) szinten is elérhető. Tartalmaz egy AdLib-kompatibilis FM-szintetizátort, mintavételezőt és sok programot is. A programok közül alapvető a Windows-alkalmazási felület, ezenkívül az SB Talker (szövegolvasó program, amely bármilyen ASCII-szöveget felolvas — angol fonetikával), az FM-orgona (akár zenekari hangzásokhoz is), amelyhez programozható ritmus- és hangszeres kíséret is adható. A Talking Parrot (beszélő papagáj) beszédet utánoz, még sértő megjegyzéseket is tud tenni, viszont ehhez minimum 286-os alaplap kell. A Voxkit tartalmaz digitális hangú felvevő, tároló és lejátszó programmodulokat is. Gyenge zajszűrése miatt azonban saját szoftvereivel zenei alkalmazásokra kevésbé alkalmas.

A Sound Blaster Pro egy több szempontból is továbbfejlesztett változat. FM-szintetizátora már 4 operátoros, ami a bonyolultabb hangszínek, kellemesebb hangzások programozását teszi lehetővé. A hangforrások száma 20-ra bővült. A mintavételezés már sztereóban is lehetséges vele, és a mintavételezési frekvencia a CD-szabványban megkívánt 44,1 kHz-re képes. Kimenetén programozható sztereó D/A keverő is helyet kapott, és egy CD-ROM meghajtó segíti a hangforrások alkalmazhatósági körét.



A Sound Blaster 16 (nevéhez méltóan) már 16 biten kezeli a hangkimenetet és -bemenetet. A tárolás helyigényének csökkentése érdekében egyszerre történik a mintavétel sűrített módú tárolása és kibontása a felvétellel, illetve lejátszással együtt. A keverőben 32 fokozatban 2 dB-es lépésekben szoftverből szabályozható a csatornák szintje, valamint a magas és mély hangszínek is. Az FM-szintetizátor 20 négyoperátoros sztereó hangforrást biztosít. A sűrítés azonban jól hallhatóan rontja a hangminőséget. Mindegyik kártyán található egy audiokimenet — fülhallgató, hangszóró vagy erősítő számára.

Zenei „agy"

A Pro Audio Spectrum 16 saját kategóriájában a legolcsóbb kártya. Félprofesszionális célokra is alkalmas, 16 bites. Installálása csak Windows-környezetben lehetséges. Programjai (Pocket Recorder és Mixer) gondoskodnak a digitális hangfelvevő, a visszajátszó modul és a bemeneti lehetőségek (saját szintetizátor, CD-ROM, mikrofon) gyors hozzáférhetőségéről és kapcsolatáról. Sztereó felvételeket tud készíteni, editálni és visszajátszani. Mintavételezési frekvenciája 44,1 kHz-ig szabályozható, a felvett hang megfelelő felvételi szint esetén CD-minőségű. Szintetizátorának hangja a legtöbb esetben kiváló, dobszekciója is kielégítő. A zenei editor programja hagy kívánnivalókat maga után (egy régi, nem túl felhasználóbarát Voyetra-sequencer). Saját vagy külső hangmintákat használható effektekkel tudja ellátni (visszhang, termesítés stb.), és visszafelé is el tudja játszani a dallamokat. Aki hobbiszinten igényes, ezzel a kártyával sok mindent meg tud valósítani.

A MIDI Blaster tulajdonképpen nem kártya, hanem a népszerű nevén „agy"-nak becézett eszközök (billentyűzet nélküli szintetizátorok) kategóriájába tartozik. A számítógéppel vagy más szintetizátorral a szabványos General MIDI-interfészen keresztül vezérelhető. 20 polifonikus hangzást és ütősök hangját szintetizálja a PCM (saját ROM-ban tárolt hangzatok) és FM szintetizálási technika együttes alkalmazásával. Előre programozott hangzásainak száma 128, de emellett 33 hangeffektus és 55 ütőshangzat is lekérhető róla.

A Port Blaster hangmodul szintén külső egység. Azáltal más, hogy nem a PC buszáról vagy MIDI-interfészről, hanem a párhuzamos (printer-)portról működik. Így gyakorlatilag minden közönséges PC tulajdonosának rendelkezésére áll mint polifón külső hangmodul.

Az MT-32 ugyancsak egy „agy", és a játékszoftverek nagy része ismeri. Ez sem az elmúlt napok terméke, de hangcsaládja egy amatőr szintű, zenekedvelő PC-st minden szempontból kielégíthet. Szintetizátora megfelelő szoftverrel kellemes hatású zenék komponálására (és meghallgatására) alkalmas.

A Roland LAPC-1-et az MT-32 kártyaformátumú változatának is nevezhetnénk, bár jóval igényesebb célokra is használható. 32 hangig polifonikus, 8 szintig multifón, és nagyszámú ütőhangszer hangja nyerhető ki belőle professzionális minőségben. Zengetőjével együtt mindenképpen a legjobb hangzású kártyák közé sorolható, így nemcsak játékokhoz és kísérőzenékhez jó, hanem még zenészeknek is. (MIDI csatlakozózsinór, valamint az erősítőkimenethez kellő jackdugó jár hozzá.) A különálló, ám nem szükséges MIDI csatlakozódobozt vagy csatlakoztatható „buta" billentyűzetet — hacsak nincs már saját MIDI-interfészes szintetizátorunk — külön kell hozzá megvásárolnunk, amennyiben hangszer használatára is vágyunk.

Pintér István—Törley Dezső

Zenekar a dobozban

Te lehetsz a szólista!

Nemrégiben egy érdekesnek ígérkező program „érkezett" a frankfurti Music Messe-n járt látogatóktól. A program neve „Band in a Box", azaz zenekar egy dobozban. A programnak két változata van, egy DOS-os és egy windowsos. Ez utóbbit vette a szerző szemügyre.

Első látásra a program közönséges sequencernek tűnik, azaz olyan programnak, amellyel zenét lehet írni, az a dallamokat visszajátssza, s komponálható mellé valami kíséret is — mintha többsávos magnó volna. Azért a sequencerek és a többsávos magnók között van némi differencia: az utóbbiak a teljes zeneanyagot rögzítik, a sequencerek csak a MIDI vezérlőjeleket, az ún. MIDI-eseményeket.

Persze a programhoz kell valami hangkártya is, hogy „jól szóljon", megfelel hozzá bármi, aminek van Windows multimédia driver programja. Például egy SoundBlaster, egy AdLib, vagy éppenséggel egy Roland SCC-1. Én ez utóbbival próbáltam ki a programot, mondhatom csodás hangokat sikerült előcsalogatni belőle.

Hogy a program mégsem egy „közönséges" sequencer, arra csak akkor jöttem rá, amikor már használtam. Tudniillik a Band in a Box nem a minél tökéletesebb sequencer szolgáltatásokra van kiélezve, hanem arra, hogy a zenei stílusok egyszerűen bevihetők, illetve elérhetők legyenek. Gyakorlatilag egy szalonzenekart építettek be a programba.

A programhoz ezer meg ezer demo van, stílusok és dallammal is rendelkező zenedarabok. Bármelyik zenedarabot bármelyik stílussal le lehet játszani. Itt van rögtön az első demo, az Alexander's Ragtime Band. Amikor beolvassuk, alapvetően a megfelelő Jazz Swing 4/4-es stílus jön be vele, ha akarjuk. Ez azonban megváltoztatható countryra, funkra, vagy éppenséggel heavy metalra. Egészen furcsa hatások csiholhatók ki a gépből, illetve a hangkártyából — egyszerűen a stílusváltással. Természetesen új stílusoknak is szabad az út!

Érdekessége a programnak, hogy a szám lejátszása közben mutatja a zongorán lefogandó akkordokat (egy szimbolikus zongorabillentyűzet segítségével), a gitárakkordokat a megfelelő szöveges jelekkel, sőt, ha a számban még eredetileg ének is lenne (merthogy a hangkártyák sok mindent tudnak, de énekelni azért nem), szóval a szöveget is mutatja a program a megfelelő ütem mellett. Ha az ember megfelelően hangszigetelt helyiségben tartózkodik, ahol a család többi tagja nem hallja, még azzal is megpróbálkozhat tehát, hogy szólót énekel a dallam fölé.

A programot legjobban tánciskolák, bárzenészek, vendéglátósok, szalonzenekarok használhatnák ki a gyakorláshoz, vagy olyankor, ha a zenészek nincsenek elegen. Sőt, tudnám ajánlani zenetanuláshoz, összhangzattani gyakorlatokhoz is, mert meglepően gyorsan rá lehet hangolódni az egyes stílusokra, és megpróbálhatunk játszani valamit az adott „hangszereléshez".

A programnak egyetlen szépséghibáját fedeztem fel, mégpedig azt, hogy alapértelmezésben a behívott számot a program végtelen ciklusban játssza le, bár ez a mód is kikapcsolható. Kissé idegesítő lehet egy zeneszámnak mondjuk hetvennyolcadszori, egyvégtében történő meghallgatása.

Sík Zoltán

Csudára megbízható!

A „jó öreg" új köntösben

Az Alaplap szerkesztősége egy emberként esküszik a régi Keditre (mint „mezei" szövegszerkesztőre, DTP-előkészítőre, „rendezőpályaudvarra"). Pedig elsősorban programozók, fejlesztők számára készített editor, s direkt szövegformázási lehetőség csak kevés van benne, de egyszerűbb levelek, dokumentációk írására „lebutítva" is elegendő.
Kíváncsian vártuk hát az új verzióról (v5.00D1) szóló híreket!

A program „installálása" egyszerűen a megfelelő fájlok bemásolásával történik, helyigénye mindössze 300 kbájt, még az online helppel együtt is csak 850 kbájt.

A Keditet parancssorból, illetve a billentyűzetről lehet vezérelni. Menü nincs, ez a megközelítés is a program szemléletéből származik: elsősorban a hatékonyságot és a gyorsaságot tartja szem előtt.

Egyszerre legfeljebb 20 (40?) fájlt lehet kezelni, az összméretet, illetve az egyes fájlok méretét csak a rendelkezésre álló memória korlátozza. A program egyaránt tud használni EMS-t, XMS-t és UMB-t. Emiatt már 286-os gépen is lehet igen nagy fájlokkal dolgozni.

Maximum 4x2 ablakban lehet több fájlt vagy egy fájl különböző részeit egyidejűleg szerkeszteni. Nincs átlapolás, és az ablak szélessége csak 40 vagy 80 karakter lehet. A Kedit a videokártyától függően képes 28, 43, 50 soros módban is működni.

Háromféle blokk van: sor-, téglalap- és folyamszerű. Egyszerre egy blokk lehet kijelölve, ez törölhető, másolható, mozgatható, sokszorozható, illetve speciális esetekben felül lehet vele írni egy részt, illetve fel lehet tölteni egy adott karakterrel. A blokkműveletek egyszerűek, és ablakok, illetve fájlok között is működnek.

A program támogatja az egér használatát. Pozícióváltásra, blokk kijelölésére stb. használható az egér, nagyon intuitív módon. Lehetőség van egérvezérelt „toolbar" kialakítására is.

A Kedit teljesen a felhasználó ízlése szerint konfigurálható, ebbe a színektől kezdve az egyes billentyűk kiosztásáig minden beleértendő. Akárhány konfigurációs állomány lehet, ezek közül a megfelelő kiválasztása akár manuálisan (parancssorból), akár path-keresési logikával történhet.

A keresési és változtatási műveletek széles skálája áll a felhasználó rendelkezésére; az egyetlen hiányosság az, hogy a reguláris kifejezéseknek csak egy szűk része van implementálva. Valamennyi keresési és változtatási művelet korlátozható: csak a kijelölt blokkban, akár a sorok egy részén vagy megadott oszloptartományban hajtódik végre.

Ez az új változat tartalmaz undo és redo lehetőséget, ennek mértéke ugyanúgy beállítható, mint minden másé. Az undo és a redo sorszintű, ezzel is javul a hatékonyság.

A Keditnek egy nagyon ügyes, KEXX nevű makrónyelve van. Ez a többi editortól eltérően nem a C szintaxisát követi, hanem az IBM nagygépeiről ismert REXX nyelv egy megvalósítása. A vezérlési struktúrák garmadáján kívül minden editorfunkció elérhető belőle, különleges változótípusok segítik a speciális editálási feladatok megoldását. A makrók csak külső fájlból hívhatók be, viszont — és ez is új ebben a verzióban — kiváló nyomkövetés van.

A KEXX makrók és a Kedit egyéb szerkesztési lehetőségeinek felhasználásával sorszerkezetű ASCII állományokon komoly adatmanipulációt és formázást lehet végezni, a rendezéstől az oszlopösszegek képzésén át akár egyszerűbb felhasználóbarát lekérdezőprogramok készítéséig.

A programban van néhány különleges lehetőség, ami sehol máshol nem található meg. Ezek közül ki kell emelni a szelektív szerkesztést. Ilyenkor csak azok a sorok láthatók és szerkeszthetők, amelyek egy bizonyos — akár nagyon bonyolult, továbbá makróból is szabályozható — feltételnek eleget tesznek. Így például könnyű egy programban csak egy adott nevű változót tartalmazó sorokkal foglalkozni.

Néhány további különlegesség: mód van Unix-formátumú, azaz csak soremelés karakterrel lezárt sorokból álló állományok olvasására és írására. Ez a mindkét operációs rendszerben dolgozók számára nélkülözhetetlen. A Kedit hálózatban is működik, ekkor megadható szabályokkal zárolja a használt fájlokat.

Fontos még tudni, hogy a Kedit kiemelkedően gyors, több Mbájtos fájlokon is másodpercek alatt keres, változtat. A program tökéletesen megbízható, 8 évi használat alatt egyszer sem volt hiba, egyszer sem szállt el, s a kézikönyvtől eltérő működést sem produkált. (Bár mi még csak hat éve használjuk, tapasztalatunk ugyanaz. *A szerk.*)

A Mansfield Group legújabb Kedit-verziójának (v5.00D1) ára 19 000 Ft.

Horlai János

```
c:kedit.txt    line=0      column=1    size=94     memory=361K  ' '=20/032    i
 5 occurrences changed in 3 lines
     |...+....1....+....2....+....3....+....4....+....5....+....6....+....7.]..
              T                              T                    T

00000 * * * top of file * * *
00001 @20 HV NAR BAL = A "jó öreg" új köntösben
00002
00003 @40 HV NAR BAL = Csudára megbízható!
00004
00005 @LEAD = Az Alaplap szerkesztôsége egy emberként esküszik a régi Keditre
00006 (mint "mezei" szövegszerkesztôre, DTP-elôkészítôre,
00007 └rendezôpályaudvarra¬). Pedig elsôsorban programozók, fejlesztôk számára
00008 készített editor, s direkt szövegformázási lehetôség csak kevés van
00009 benne, de egyszerûbb levelek, dokumentációk írására "lebutítva" is
00010 elegendô. Kíváncsian vártuk hát az új verzióról (v5.00D1) szóló híreket!
00011
00012 A program "installálása" egyszerûen a megfelelô fájlok bemásolásával
00013 történik, helyigénye mindössze 300 kbájt, még az online helppel együtt
00014 is csak 850 kbájt.
00015
00016 A Keditet parancssorból, illetve a billentyûzetrôl lehet vezérelni. Menü
00017 nincs, ez a megközelítés is a program szemléletébôl származik:
00018 elsôsorban a hatékonyságot és a gyorsaságot tartja szem elôtt.
F1:Help  F2:Add  F3:Quit  F4:Keres  F5:Cu=CL  F6:?  F7:Nextw  F8:Dup  F9:VP-Help
```

Ezt a cikket ezért még a 3.51-sel szerkesztettük...

WinMaster és környéke

Ügyesebben, sebesebben!

A Windowsban való munkát segítő segédprogramoknak se szeri, se száma. Azonban kevés olyan van, amelyik átgondoltan tartalmazná a szükséges elemeket: nem többet és nem kevesebbet. Ezen programok közül is kiemelkedik a WinMaster, a PC-Kwik cég terméke.

A program két lemezen van, installálása könnyű, de kér egy sériaszámot, ezzel nehezítve (egy keveset) a szoftverkalózok dolgát. A teljes program mintegy 3 Mbájtot foglal el a merevlemezen, ez a rengeteg szolgáltatáshoz képest szerénynek mondható, és — szokatlan és előremutató módon — a program nemcsak telepíthető, hanem ki is telepíthető (deinstallálható).

Kezdetben minden windowsos képernyő szinte üres, de a programok számával elkezdenek szaporodni az ablakok, amelyek előbb-utóbb még a legnagyobb (és olvashatatlan) felbontás mellett is eltakarják egymást, sokszor nem könnyű az éppen általunk keresett ablakot megtalálni, és abban a megfelelő programra rákattintani. Ezen a problémán segít a Toolbox.

Tetszőleges számú, tartalmú, elrendezésű és elhelyezkedésű „dobozt" definiálhatunk, ezekben a gyakran használandó programok ikonjai vannak, amelyek (programcsoporttól függetlenül) egyetlen kattintással indíthatók. Még azt is eldönthetjük, hogy egy-egy ilyen dobozt eltakarja-e a többi ablak, vagy pedig mindig látszódjék.

Választhatunk, hogy kisebb vagy nagyobb méretben legyenek benne az ikonok. Sőt, bármelyik gomb egy beágyazott doboz indítógombja is lehet. A dobozok szerkesztése egyszerű, a szükséges program ikonját akár az egérrel is bevonszolhatjuk a helyére.

```
              PC-Kwik(R) Memory Map Utility, Version 1.20
       Copyright 1990 - 1992 PC-Kwik Corporation.  All Rights Reserved.

                    DOS Memory Usage -- DOS Version 5.0

                      DOS Conventional Memory Usage

    Size      PSP  Name              Command Line
   44,176     0008  Kernel
    3,504     0AC9  Shell
      432     0BA9  Not available   /lh /chkhi /copy /windows
    1,584     0BCF  Shell
   11,760     0C34  win386.exe        r t    w 2  2        e T 3
    3,616     0F0D  Shell
  590,288     Free.  Largest free block is 557,856.
  -------
  655,360     Total Conventional DOS memory.

                         DOS Upper Memory Usage

    Size      PSP  Name              Command Line
  167,504     0008  Kernel
   38,592     0BA9  Not available   /lh /chkhi /copy /windows
Press any key to continue...
```

Egy másik PC-Kwik program képernyője — még DOS környezetben

A Windows működési sebességét alapvetően meghatározza, hogy milyen gyakran kell a lemezhez fordulnia. Ezen segítenek a disk-cache programok, melyek a memóriában tartják a lemez tartalmának egy részét, és amikor egy alkalmazásnak szüksége van rá, akkor onnan bocsátják a rendelkezésére. A Windowshoz a Microsoft adja a Smartdrive cache programot, amely (a korábbi verzióihoz képest) nagyon sokat fejlődött. De minél jobb, gyorsabb cache programunk van, annál kevesebb a munka közbeni fölösleges várakozás.

Memóriakölcsönző

A PC-Kwik Superwinje igen jó program. Különleges szolgáltatása az ún. kölcsönzés, ami azt jelenti, hogy a cache egy általunk meghatározott hányada dinamikusan visszaadódik az éppen futó alkalmazásoknak, ha több memória kell hozzá. Amikor ezeknek már nincs szükségük a memóriára, akkor a Superwin újból a saját céljaira használja fel.

A setup műveletek során mindent beállíthatunk: a cache méretét, azt, hogy mely drive-ok tartozzanak a cache hatása alá, és melyek ne, hogy a cache program kevesebb helyet foglaljon el, de lassúbb legyen, vagy éppen fordítva. Megadhatjuk, hogy az olvasásnak prioritása legyen az írással szemben, hogy az írási igényeket egy darabig gyűjtse a program, és azután rendezetten, a fej mozgását követve írja a lemezre. Beállítható egy várakozási idő is, amíg az írási igények összegyűlnek. Ennek megvan az a kockázata, hogy éppen ilyenkor száll el valami, és a ki nem írt adatok elvesznek. De ha célszerűen kicsire állítjuk ezt az időt, akkor ez az adatvesztési veszély minimálisra csökken, míg a sebesség tényleg szembetűnően nő. (A kézikönyv a 2 s-ot javasolja, én a tapasztalatok alapján, elég nagy memória mellett, inkább az 5-öt tartom jónak.)

A disk-cache mellett egy speciális RAM-disk driver is tartozik a WinMasterhez. Ennek az előzőekkel összefüggő specialitása, hogy mérete dinamikusan változhat, a cache és a RAM-disk osztoznak a memórián. Tehát ha erre a virtuális diszkre több fájlt írunk ki, akkor mérete nő, miközben a cache mérete csökken, törlés esetén PEDIG éppen a fordított jelenség játszódik le.

Tapasztalatátadás

Meg kell azonban jegyezni, hogy cache-szoftverek használata mindig kockázatos! Ezek általában nagyon „mélyen" nyúlnak bele a gépbe, és még a nagyon jól megírt cache is okozhat rendszerösszeomlást. Mintegy kéthetes tesztelés során ez velem is megtörtént néhányszor, főleg ha a DDE lehetőségeit használtam. Mióta azonban a STACKS= sort 20,512-re állítottam be, azóta ilyen probléma nem lépett fel. A másik érdekesség (ez DOS-ból tapasztalható inkább): akár egy kisméretű programot is háromszor kell elindítani ahhoz, hogy utána már ne legyen lemezművelet. (A józan ész szerint már a második indításnál is mindennek a cache-ben kellene lennie.) Hiányzik a sok más hasonló programban meglévő lehetőség, hogy menet közben valamilyen billentyűkombinációval ki-be lehessen kapcsolni a cache működését. Ez különösen rendszerhibák esetén volna nagyon hasznos.

Bár a Superwin beállítása a Setup programból igen könnyű és elegáns, mind a help, mind a kézikönyv keveset mond a különböző lehetőségek pontos hatásairól. Ráadásul például az SCSI-meghajtókkal kapcsolatban javasol egy opciót (a Smartdrive /double-bufferjének megfelelőt), de nem ír arról, hogy mi a teendő, ha egy gépben kétféle kontroller is van. Általában, bár az átlagos felhasználónak valószínűleg ennyi információ is sok, a téma érzékenységére való tekintettel kifejezetten keveslem a kézikönyv információit.

Információ — jó, ha van

A WinMaster tartalmaz egy sor információs programot is. Ezek egy része statikus, a rendszerrel, az éppen futó programokkal, memóriával kapcsolatos adatokat jelenít meg. Másik részük azonban rendkívül hasznos lehet, ezek dinamikusan figyelik a CPU-t, a rendszer erőforrásait, a memóriát, a lemezterületet és a disk-cache-t. Adataikat idősoros grafikonban nézhetjük, illetve ikonizálva, mint kis mérőműszer mutatják az aktuális értékeket. Ilyen módon követni lehet, hogy egyes programok hogyan használják a rendszert, mit tegyünk, hogy optimálisabban használhassuk a Windowst.

Valamennyi műszer igen jól paraméterezhető, a gyűjtési idő hosszától kezdve a grafikon típusáig minden lényeges dolgot beállíthatunk. A WinMasterhez tartozik egy sor segédprogram is, mely a lemezt vizsgálja, annak működését optimalizálja.

Megnézhetjük, hogy melyik drive-on mennyi szabad hely van (ezt könyvtárakra, alkönyvtárakra lebontva is vizsgálhatjuk), a megjelenítés szép és informatív, lényegesen többet mond, mint a dir vagy akár a File Manager.

Vizsgálhatjuk a fájlok fragmentáltságát, az ezt mutató program különösen hasznos, és lényegesen több információt szolgáltat, mint a Norton Speeddisk. Bármelyik szektor tartalma megtekinthető, megnézhetjük, hogy egyes fájlok fizikailag hogyan helyezkednek el a lemezen, illetve a lemez mely része melyik fájlhoz tartozik.

Egy jól használható, aprólékosan beállítható defragmentáló program egészíti ki ezt a szolgáltatást. Ennek külön előnye, hogy időzítetten indítható, akár a Windows minden indulásánál, vagy bizonyos napokon ellenőrzi a megadott logikai meghajtókat, és a beállított értékektől függően automatikusan elindítja a defragmentáló programot.

Inkompatibilis archiválás

A WinMasterben van még egy hasznos archiváló program is, a KwikVault. Indítása után ikonként van jelen, és a File Managerben kijelölt fájlokat, könyvtárakat — az egérrel az ikonra mozgatva, ha akarjuk sűrítve — archiválhatjuk azt, amire éppen nincs szükségünk. Sajnos az archiválás nem kompatibilis semmilyen elterjedt programmal. Az archivált fájlok adatai a KwikVaultból megnézhetők, és ha újra kell valami, azt gyorsan és egyszerűen kiválaszthatjuk, visszatölthetjük.

Összefoglalva, a WinMaster egy olyan programcsomag, amely mindenkinek nyújt valamit, szolgáltatásai gazdagok, hasznosak és jól átgondoltak, ára nem lehet akadálya a beszerzésnek.

DOS-hívőknek

A cég, amely a WinMastert készítette, forgalmazza régóta meglévő DOS-os csomagjának új változatát is. Ez a Pc-Kwik PowerPak 3.0.

Ebben többek között a WinMaster kapcsán ismertetett disk-cache, RAM-disk és lemezvizsgáló, optimalizáló programok DOS-változata található meg. Sajnos a kézikönyv gyengeségéről mondottak itt is érvényesek. Különösen kellemetlen, hogy nincs olyan konfiguráló program benne, amely a windowsos változathoz hasonlóan kényelmessé tenné a paraméterek beállítását.

Ez kifejezetten súlyos hiányosság, a felhasználók döntő többségétől nem lehet elvárni, hogy komplikált szintaxisú konfigurációs fájlt szerkesszen valamilyen ASCII-szövegszerkesztővel.

A PowerPak a fentieken kívül tartalmaz egy printspoolert (azaz egy olyan programot, amely a nyomtatási igényeket összegyűjti, és a gép sebességének jelentős csökkenése nélkül, egyenként küldi ki a nyomtatóra). Ez a spooler is osztozik a memórián a cache és a RAM-disk programokkal. Ott, ahol sokat nyomtatnak, használata komoly időnyereséggel jár. Konfigurálására sajnos szintén érvényesek a korábban mondottak.

Gépelésben bajnokoknak

A csomagban van egy billentyűzetgyorsító és -beállító is. Ennek hasznossága felől kételyeim vannak, nagyon kevés ember gépel gyorsabban, mint ahogy azt a számítógép fogadni tudja.

Ráadásul több mai BIOS és a Norton Control Center is lehetővé teszi ezeknek a paramétereknek az állítását. A DOS-ban megadott 15 karakteres billentyűzetpuffer mérete is növelhető, ez néha tényleg hasznos lehet, ha egy általunk nagyon jól ismert programban túl gyorsan gépelünk.

A fentebbi programot egészíti ki egy képernyőgyorsító. Ez fölpörgeti a text módban való kiírást és görgetést, hatásossága inkább a régi, lassúbb CGA-kártyáknál érvényesül. Viszont kellemes szolgáltatása, hogy a képernyő tetején „kicsorgott" sorokat újra megnézhetjük, és ez sokszor nagyon jól jön.

A két fenti programmal együtt működhet egy ReDos nevű parancsszerkesztő, amely kb. ugyanannyit tud, mint a Doskey, és sokkal kevesebbet, mint a 4DOS.

Összefoglalva, a PowerPak csomagból a lemezkezelő rész jó, különösen a PrintSpooler — DOS-os felhasználók komoly hasznát vehetik. A többi elem jórészt már idejét múlta, inkább csak hagyományból maradhatott a csomagban. Másrészt, ha figyelembe vesszük, hogy rengeteg AT és XT is forgalomban van még, akkor a képernyővel, billentyűzettel kapcsolatos gyorsító programoknak is megvan a jogosultságuk.

Horlai János

SolarSoft



Megrendelem postai utánvéttel az alábbi SolarSoft lemezeket. A vételárat és a postaköltséget átvételkor fizetem.

000 Katalóguslemez (2 lemezen, 1993. januári)	..... példány
467 Amy's First Primer	..... példány
471 Spellbound	..... példány
629 Miramar	..... példány
630 Display Font Editor	..... példány
632 Hyperdisk	..... példány
636 School-Mom 3.56	..... példány
655 SoundBox	..... példány
656 Instrumenten Editor	..... példány
666 Finnish	..... példány
667 Spanish	..... példány
676 Word Gallery	..... példány
597 Viccek	..... példány
591 Commander Keen I.	..... példány
588 Geoclock & Globe	..... példány
568 Cápák és Szuperlégy	..... példány
589 Moraff's Word	..... példány
567 Vampyr	..... példány
580 Mercury	..... példány
592 Commander Keen IV. (2 lemez)	..... példány
595 Duke Nukem	..... példány
587 Nervous System	..... példány

Lemezárak:	Nettó ár	(Bruttó ár)
1 lemez	399 Ft	(499 Ft)
5 lemeztől	379 Ft/db	(474 Ft)
10 lemeztől	359 Ft/db	(449 Ft)
25 lemeztől	339 Ft/db	(424 Ft)
KATALÓGUSLEMEZ CSAK	199 Ft!	(249 Ft) (2 lemezes!)

INFORMÁCIÓKÉRÉS: A0307 ▶

Előfizetés az Alaplapra

Előfizetek az Alaplap című, havonta megjelenő számítástechnikai folyóiratra .... példányban, ❑ 1 évre, ❑ fél évre.

❑ Az előfizetési díjat számlájuk alapján **átutalással** egyenlítem ki.

❑ Kérem, hogy az előfizetési díj befizetéséhez küldjenek **csekket**.

Név: ....

Cím: ....

Helység, irányítószám: ....

Dátum: Aláírás:





MEGRENDELŐLAP

Megrendelem utánvétellel az Alaplap kiadványsorozataiban megjelent alábbi műveket:

ALAPLAP KÖNYVEK

... pld: Nagy Gábor: Tömör gyönyör	156,–
... pld: Kis János–Szegedi Imre: Vírushatározó	256,–
... pld: Jodál Endre: Általános fogalmak (Számítástechnikai alaplexikon I., 2. kiadás)	356,–
... pld: Jodál Endre: Adatkommunikáció és számítógép-hálózatok (Számítástechnikai alaplexikon II.)	356,–
... pld: Farkas Ernő: PC-szótár	456,–
... pld: Kis János: BBS — avagy az elektronikus postaláda (lemezmelléklettel)	656,–
... pld: Jodál Endre: Informatikai alapszótár	356,–
... pld: Csórián Sándor: Számítógépes kommunikáció	356,–

ALAPLAP FÜZETEK

... pld: Detrik Péter: Az SQL nyelvről	375,–

ALAPLAP LEMEZEK

... pld: Bliss főkönyvi program (demó és leírás)	750,–
... pld: Norton Guide keretprogram (leírás)	500,–
... pld: PathMinder segédprogram (leírás)	500,–
... pld: CSProlog nyelv (leírás)	1000,–
... pld: LIM EMS 4.0 memóriakezelő (leírás)	1000,–
... pld: Magyar betűkészletek Windows 3.0-hoz	1000,–

INFORMÁCIÓKÉRÉS: A0565 ▶

INFORMÁCIÓKÉRÉS

Kérem, hogy az itt általam **BEKARIKÁZOTT KÓDSZÁMÚ** hirdetésekkel kapcsolatban küldjenek részemre bővebb tájékoztatást.

A0114	A0138	A0141	A0206
A0219	A0307	A0317	A0318
A0335	A0340	A0341	A0353
A0354	A0404	A0428	A0439
A0440	A0444	A0454	A0458
A0463	A0470	A0479	A0482
A0501	A0503	A0505	A0506
A0509	A0512	A0514	A0516
A0518	A0519	A0520	A0521
A0524	A0525	A0528	A0529
A0530	A0531	A0534	A0535
A0537	A0538	A0540	A0541
A0542	A0545	A0546	A0548
A0549	A0553	A0554	A0555
A0556	A0558	A0564	A0565

Beküldhető: 1993. június 30-ig

FELADÓ

A) Egyéni érdeklődő:

Név: ....

Cím: ....

Helység: ....

Irányítószám: ....

B) Vállalati érdeklődő:

Cég: ....

Ügyintéző: ....

Cím: ....

Helység: ....

Irányítószám: ....

Telefon/Fax: ....



IDG Magyarországi
Lapkiadó Kft.
Pf. 386

Budapest
1536



ALAPLAP

Ami minden PC-hez kell



IDG Magyarországi
Lapkiadó Kft.

Budapest
1536

FELADÓ:

Név: ....

Cég: ....

Utca, házszám: ....

Helység: ....

Irányítószám: ....

Telefon/Fax: ....



Cédrus Kiadó
Pf. 74

Budapest
1441

FELADÓ:

Név: ....

Cég: ....

Utca, házszám: ....

Helység: ....

Irányítószám: ....

Telefon/Fax: ....



Cédrus Informatikai Rt
Pf. 71

Budapest
1251

A LEMEZMELLÉKLET TARTALMA:

- A hónap témájához: Mértékegység-átszámító
- A nyelvek kódlogikája
- DES-titkosító — forráskódban
- A hibajavító kód (Nagy Sándor)
- Függvénymódosítások a Clipperhez (Fürstner János)
- Egy lemezt — egy fájlba (Cseppentő Árpád)
- Táblázatokat generáló program (Jánosi Tibor)
- Lemezellenőr — a Disktool után (Farmosi István)
- Egy májusi vírus, az SWB ellenszere (Farmosi István)
- Snoboláljunk (Vargha Dénes)
- Játék — ezúttal Windows alatt





KAO — a tökéletes memória

A határ a csillagos ég!

Manapság a PC-s programok között egyre nagyobb számban találni olyat, ami oktató-tanító jellegűnek tekinthető. Az Easy Cosmos for Windows egy csodálatos, csillagtérkép-ábrázoló és -vizsgáló program, amelyet egyaránt élvezhet és használhat az érdeklődő általános iskolás és az amatőr csillagász. Használatához Windows, 6,5 Mbájt szabad lemezterület és VGA-kártya szükséges.

Indítás után a csillagos ég színes képét látjuk. Be lehet állítani, hogy honnan nézzük az eget. Budapestről, de akár a déli féltekéről látható égbolt képe vizsgálható. Rengeteg, előre definiált amerikai és egyéb helyszín adata van meg, de mi is megadhatunk újat. Az egér segítségével mozgathatjuk az égbolt képét az időben, így végignézhető egy este kozmikus képe, de megnézhetjük, hogy milyennek látták őseink, vagy milyennek fogják látni utódaink.

Ki lehet nagyítani egy tetszőleges részt, és alaposabban megvizsgálni, majd egy mozdulattal újra visszatérhetünk a teljes égbolt képéhez. Bejelöltethetjük a csillagképek képzeletbeli meghatározó vonalait és a koordinátaköröket is. Így könnyebben megtaláljuk, amit keresünk, vagy könnyebb az összevetés másfajta csillagtérképekkel.

Bármelyik objektumra ráállhatunk az egérrel, és ha akarjuk, egy kis ablakban megjelenik a neve, adatlapja, de megtehetjük azt is, hogy ez az alakzat kerüljön a kép középpontjába. Ha egymáshoz túl közel eső csillagok közül kattintunk rá valamelyikre, akkor sorban végignézhetjük őket. Számos égi objektumról (főleg az érdekesebbekről) gyönyörű színes képet is tartalmaz a program, ez is megnézhető.

Ha tudjuk a nevét annak, amit keresünk, egy egyszerű párbeszédes ablakban megadhatjuk, és a képen rögtön látszik a helye. A névmegadásnál számos, a csillagászatban szokásos konvenció (NGC, SAO) alapján dolgozhatunk.

Szabályozható, hogy milyen típusú égi alakzatokat lássunk (csillagok, ködök, bolygók stb.). Be lehet állítani azt is, hogy milyen magnitúdójú csillagok látszódjanak. Aki este az „igazival" akarja összevetni a képen látottakat, az beállíthatja, hogy a kép a szemet kímélő vöröses színben jelenjen meg. A program még megfelelő kiegészítéssel ellátott teleszkóp hozzákapcsolásához is lehetőséget ad.

Sokféle segédprogram egészíti ki az alapterméket. Ki lehet rajzolni a bolygók mozgását, a Hold, illetve a Jupiter holdjainak pályáját, és ilyenkor nemcsak azt lehet beállítani, hogy milyen időintervallummal lépkedjen a program, de még azt is, hogy a mozgás nyoma látszódjon-e vagy sem. Megkereshetjük a bolygók együttállását, valamint a hold- és napfogyatkozások idejét is. A program mintegy 7000 forintba kerül.

H. J.

A TelePad SL tollgép

Munka a „pen-Windows" alatt

Nálunk itthon még eléggé furcsaságszámba megy egy olyan számítógép, amelyen nincs billentyűzet. A pen-computereken márpedig nincs, még véletlenül sem. Helyette egy tollat kap az ember, amellyel egyszerűen a képernyőre ír.
Végül is ez valamivel természetesebb, mint az adatok bevitele billentyűzettel, illetve a grafikus adatoké egérrel, és még gépelni sem kell tudni hozzá.

Méltán tarthat érdeklődésre számot a tollas eszköz azok körében, akik alapvetően félnek a számítógéptől, hátha egy „rossz gomb megnyomásával még valami bajt csinálnak". Itt tehát ilyen nem fordulhat elő. Sőt! Még a grafikus adatokat is ugyanazzal az eszközzel, az iskolában már jól megszokott tollal lehet bevinni a gépbe. Még a menüpontok kijelölésére is ugyanez az eszköz használható. De lássuk közelebbről a gépet!

A TelePad amerikai cég egészen friss terméke a TelePad SL pen-computer. (Gyanúm szerint a cég fő, sőt talán egyetlen profilja a pen-computerek előállítása.) A gép Intel 386SL processzorra épül, amely 25 MHz-es órajellel működik. Memóriája 20 Mbájtig bővíthető (hogy hogyan, azt nem tudom, mert nem szedtem szét), de alapkiépítésben 4 Mbájttal szállítják. Így a Windows el tud indulni rajta 386 enhanced módban is. A Windowsra pedig nagy szüksége van a gépnek, illetve annak, aki „firkálni" szeret, mivel a géphez kapcsolódó legtöbb applikáció Windows-alapú. Bár a leírás úgy fogalmaz, hogy a gép működik minden „tollas" operációs rendszerrel. (Meg kell mondjam, én még csak Windowsszal láttam pen-computert működni, már csak azért is, mert van hozzájuk „pen-Windows").

Bárhol, bármilyen helyzetben

Természetesen a Windowshoz megfelelő grafikus képernyő is kell; a TelePad cég a gépet VGA LCD háttérvilágítós kivitelben szállítja. Ez nem is érdemel figyelmet, azonban a pen-computerek esetében az adatbevitel helye is a „display". Persze ez így nem igaz, de a felhasználó kb. ezt hiszi, amikor a képernyőre ír. Ezeknél a gépeknél tudniillik a képernyő egyben tabletként is szolgál.

Az általam kipróbált gép winchestere 120 Mbájt körüli volt; maximum 250 Mbájtos lehet. A gép beépített fax- és adatmodemmel rendelkezik, a faxmodem természetesen 9600 baud szinkron üzemmódú, az adatmodem rész pedig a szokásos 2400 baudos sebességű. A géphez opcionálisan kapható külső floppymeghajtó is, további akkucsomag, digitális kamerascanner (!), koprocesszor, laptop printer, zsinór nélküli (!) hálózati csatolókártya, CD-ROM olvasó, rádiótelefon-interfész (!), vonalkódolvasó, sőt a régi bevált módszerekhez ragaszkodó felhasználók kedvéért még billentyűzet is.

Ezek után nézzük, hogyan használható a gép. Először is kézbe véve, lehet rá „jegyzetelni", vagy fogantyúját kihajtva asztalra lehet állítani, mint egy családi fényképet (kb. olyan lapos is, mint egy képkeret). A pen, amellyel írunk rá, valójában egy kicsit vastag toll, azzal a különbséggel, hogy nem fog, és a hegye benyomódik, ha erősebben megnyomjuk. Ez a benyomott hegy felel meg az egérbillentyű lenyomásának. Azért nem igazán komfortos ez a megoldás, mivel az ember általában nem a papír síkjára merőlegesen tartja a tollat írás közben, hanem döntve. Megdöntött helyzetben viszont elég nehéz benyomni a pen hegyét. Talán, ha valami golyóstollhoz hasonló megoldást találnának...

A balkezesekre is gondoltak

A Windowsba behatolva a pennel egész kényelmesnek tűnik a használat, csak rá kell mutatni a megfelelő „gomb"-ra, és megnyomni a toll hegyét (click!). Engem azért — balkezes lévén — egy kicsit zavart, hogy a kurzor mindig ott lábatlankodott a képernyőn, ahol a pen hegye tartott, csakhogy az jobbról eltakarta azt, amit éppen ki akartam választani, balról pedig a pen takarta el. Mígnem rájöttem, hogy ezt a speciális „pen-Windows"-t lehet balkezesre is konfigurálni, így mindjárt kényelmesebb a helyzet. Sőt, ha kedvem van a pen-computert állított képernyőként használni, a Windowst úgy is konfigurálhatom, és ezek után minden alkalmazás az állított képernyőt használja (jóllehet ez csak képernyődriver kérdése).

Akadt még egy probléma. Mégpedig az, hogy a „pen" néha elvesztette a kurzort — vagy fordítva —, mindenesetre időnként úgy kellett a kurzort „kézen fogva" odavezetni a kiválasztandó objektumhoz.

Bemutat(koz)ások

A géphez a gyártó természetesen mellékelt egy-két bemutatószoftvert is, amelyek között egészen érdekeseket is találtam. Az egyik ilyen fogorvosi rendelőkhöz tervezett betegnyilvántartás volt, ahol az előre megrajzolt fogakon az orvos egyszerűen besatírozhatja, hogy a betegnek melyik foga lyukas és hol. Persze a pennel rajzolni a legkönnyebb, ezért ahol lehet, érdemes grafikusan ábrázolni a dolgokat, és grafikus adatbevitelre felkészülni a programmal. Sajnos a beteg nevének bevitelét grafikusan megoldani elég nehézkesnek tűnt...

Az adatlapra az orvos egyszerűen felírja a pennel a beteg nevét, a program pedig egy karakterfelismerő program segítségével megpróbálja azt karakteres információvá varázsolni. Meg kell, hogy mondjam, ez nem sikerült neki. Ennek több oka volt — először is rondán írok, de sebaj, mert megkértem a kolléganőmet, hogy írja fel a nevét az adatlapra. A program azonban a sokkal szebb női írást is csak nyomokban ismerte fel.

Azután itt volt a másik probléma, mégpedig az, hogy a karakterfelismerő „nem tudott magyarul", azaz az ékezetes betűkkel elég nagy gondban volt.

Végül is a nevem helyett — mert természetesen a saját nevem bevitelével próbálkoztam — egy betűkből és számokból álló karakterhalmazt kaptam. Az a helyzet, hogy egy billentyűzettel tizedannyi idő alatt be tudtam volna vinni a nevemet és egyéb személyi adataimat.

Más, kifejezetten jópofa demó is volt. Egy turistahajó felszolgáló pincérének adott segítséget az a program, amelyben a hajó egyik fedélzeti szintjének körvonalrajza volt látható az utasok székeivel. Ha a pincér rámutat az egyik székre, akkor megjelenik egy-két menü, ahová a pincér beírhatja a megfelelő információkat. Például: mi az utas neve, milyen ételt eszik (vegetáriánusat, nyerset, kósert, diétásat vagy normált). Tehát a pincér — azonfelül, hogy nem hoz ki húst egy vegetáriánusnak — a következő alkalommal már név szerint szólíthat mindenkit a háromszáz utas közül.

A harmadik programként egy határidőnaplót „vizsgáztattam", amelyiknek a lapjaira beírhattam az elfoglaltságaimat és egyebeket, amit csak akartam. A kép jobb oldalán volt a regiszteres rész, hasonlóan a filofaxokhoz, ahol ki tudtam választani, hogy melyik év melyik hónap melyik napjával akarok foglalkozni. Sőt, még azt is megadhattam, hogy a „papír", amelyre éppen írok, csíkos, kockás vagy franciakockás legyen.

Sokan vágyhatnak rá

További hasznos applikációkat lehet látni a gép prospektusában. Ezek között például épületek felújításához való segédprogramot, ahol az épületről készült skicc bevitele után be lehetett írni a szükséges javításokat is a hely grafikus megjelölésével. Vagy például egy autószerelő vagy egy biztosítási kárfelvevő segédprogramja, ahol az autó fotóján — amit az opcióként kapható kamerascannerrel visznek be — lehet jelölni a javítások helyét, a hozzá tartozó adatlapon pedig az egyes javításokhoz tartozó alkatrészigényt, munkaórát és a költségeket. Végül az egyik applikációban láttam egy képernyőre kirajzolt billentyűzetet, amelynek billentyűit a „pen"-nel kell megérinteni, és ettől kezdve egészen jó hatásfokkal lehet szöveges adatokat is bevinni a gépbe. (Azért tíz ujjal gépelni más, mint egy szál „ceruzával"...)

Nem folytatom tovább; az eddigiekből is látszik, hogy a pen-computerek rengeteg mindenre jók. Egy dologra azért mégsem használnám őket: ennek a cikknek a megírására.

Sík Zoltán

Mire jók a NOVRAM-ok?

Manapság egyre több elektronikai eszközben, főleg számítógépben fordulnak elő a NOVRAM-ok, azaz a nem felejtő RAM-ok (Non-Volatile RAM). Annak idején az első IBM PC/AT gépeknél kellettek volna, ha léteztek volna már...

A technika 10 évvel ezelőtti szintjén arra volt lehetőség, hogy a Nagy Kék a „setup" információkat egy statikus RAM IC-be zsúfolja (nem is kellett annyira zsúfolni azt a pár bájtot), és az egészet egy lítiumelemmel táplálta, hogy a gép kikapcsolásakor az információk nehogy elvesszenek. Aztán persze jöttek a klónmegoldások is, a négy ceruzaelemes táplálással. Már amilyen jó volt! Ha kimerültek az elemek, a tisztelt felhasználó csak leballagott a sarki trafikba, és beszerezte a következő négy elemet. (Az elemek önkisülése ez esetben egy nagyságrenddel nagyobb áramot jelent, mint a CMOS IC-k által felvett áram.) Noha ezeknek az elemeknek sem akkor, sem pedig ma nem lenne szabad lemerülniük, igenis lemerülnek. Többek szerint azért, mert — elsősorban a klón gépeknél — a géphez szállított elem enyhén szólva „állott".

Egy szó, mint száz, jobbak lennének a NOVRAM-ok. Az első NOVRAM-ot tartalmazó PC-s kártya (legalábbis ismereteim szerint) a régebben még hardverelemeket is gyártó Novell cég DCB-kártyája, azaz Disk Coprocessor Boardja volt. Itt a kártyán lógó SCSI winchesterek típusát tárolta, illetve tárolja ma is a NOVRAM. (Sőt, még legalább száz évig tárolná, ha addig nem megy ki a divatból az egész — valószínűleg winchesterestül.)

Hogyan működnek?

Az ember egyszerű aggyal arra gondol, hogy a fejlesztő urak nem tettek mást, mint egyszerűen betokoztak egy elemet a statikus RAM mellé. De nem ezt tették. Ehelyett a következőt mondták: itt van nekünk a sima statikus RAM, léteznek EPROM-ok is (ilyenben tárolja minden gép a BIOS-t) — talán jó lenne mindkettőnek az előnyös tulajdonságait egyesíteni. Az EPROM mint olyan erre nem igazán jó ugyan, mert abból a „beleégetett" információt csak UV fénnyel lehet törölni. Viszont kitalálták már az EEPROM-ot is, az elektromosan törölhető EPROM-ot is.

Ennek a működési részleteibe nem érdemes belemenni, mert gyorsan a kvantumfizika területén találjuk magunkat, ahol a legegyszerűbb kifejezés is maga a Schrödinger-egyenlet. Anélkül, hogy az ilyen káromkodásokat tovább folytatnám, elég legyen csak egy-két kulcsszó az EEPROM-ok működésével kapcsolatban, ilyenek, hogy: alagúteffektus (lásd például tunnel-dióda), vagy eltemetett réteg. Mindegy. Higgyük el, hogy az EEPROM-okat igenis lehet elektromos árammal is törölni. (Az igazság az, hogy már a mezei EPROM-ok is a fent ki nem fejtett elven működnek).

Na már most, miért jó nekünk az EEPROM? Azt azért ne gondolja senki, miszerint úgy fogjuk ezt RAM-ként használni, hogy minden pillanatban töröljük a tartalmát, majd újra-írjuk az egészet az új tartalommal. Inkább tegyük a következőt: csináljunk úgy, mint a PC-k azzal a bizonyos „shadow" RAM-mal, amelybe a gép egy lenge mozdulattal átmásolja a ROM BIOS tartalmát, hogy azután onnan gyorsabban el tudja érni, mintha közvetlenül az EPROM-ból, illetve ROM-ból futna a BIOS. Tehát tegyünk egymás „mögé" egy statikus RAM-ot, és egy ugyanakkora méretű EEPROM-ot. Ehhez már csak egy-két vezérlőjel kell, és már készen is van a NOVRAM.

De az az egy-két vezérlőjel sem akármit vezérel. Az egyik jel, illetve IC-láb a „store" (avagy tárolás), a másik láb pedig a „RECALL" (azaz visszahívás). A NOVRAM a store jelre beírja a statikus RAM tartalmát az EEPROM-ba, a RECALL jel hatására pedig vissza az egész, azaz a statikus RAM tartalmát felülírja az EEPROM tartalmával. Most már csak annyit kell az egészhez kívülről hozzátenni, hogy ha a gépünk „elveszti" a tápfeszültséget, még legyen annyi idő, hogy utolsó leheletével ki tudja adni a STORE jelet, és sikerüljön az adatok mentése. És viszont, ha megjön a tápfesz, akkor illik egy RECALL jelet kiadni, és mintha mi sem történt volna, minden megy tovább.

Ma még luxus

Miért nem használják mégsem a NOVRAM-okat például a winchesterek helyett? Pedig elérési időben sem rosszak a NOVRAM-ok, kb. 300 ns körüli értéknél vannak (egy normál dinamikus RAM elérési ideje manapság 70 ns, míg egy winchesteré kb. 15 ms).

Ennek nagyon egyszerű oka van. Ha megnézzük például az egyik vezető NOVRAM-gyártó, a Xicor cég katalógusát, meglepődve láthatjuk, hogy a NOVRAM IC-k kapacitása 64x4 bittől mintegy 512x8 bitig, azaz bájtban kifejezve 32 bájttól 512 bájtig terjed. Még jobban meglepődnénk, ha az ehhez tartozó árakat is látnánk. Azokat viszont a katalógus nem tartalmazza. Szerintem ez lehet a magyarázata annak, hogy még egy darabig winchesteren fogjuk tárolni mindenféle programjainkat és adatainkat.

S. Z.

Jobb későn...

Keresetlen szavak a Disktoolról

Az Alaplap márciusi lemezmellékletén található egy disktool.exe nevű segédprogram, amely sajnos többeknek okozott gondot az elmúlt időben. A szerkesztőség felkérésére alaposabb tesztnek vetettük alá, s ennek tapasztalatait tesszük most közzé, a program hibájából esetlegesen adódó gondok kiküszöbölésére pedig a mostani lemezmellékletre elhelyeztünk egy „Disktool utáni diszkdoktort".

A program leginkább a PC Tools, PC Shell családhoz hasonlítható segédprogram, amely különböző diszk- és fájlkezelő funkciókat tartalmaz. Megjelenése kulturált, kezelhetősége elfogadható, azonban tudásában nem nyújt többet egyetlen hasonló segédprogramnál sem, sőt: alapvető hibákat is tartalmaz. A program feltűnően ragaszkodik hozzá, szinte kiköveteli, hogy új tulajdonosa regisztráltassa, ezzel kényelmetlenné teszi kezelését. Úgy látszik, erre nagyobb gondot fordított szerzője, mint a program hibátlan működésére.

Amikor a Doctor nem segít

A tesztelést egy 286-os laptop gépen kezdtük. Itt ért az első meglepetés, mivel a program színbeállítására szolgáló menüje nem működött, szinte lehetlenné téve az LCD képernyős gépen való használatot. A többi menüpont megfelelően „hozta magát" egy kivétellel, ez pedig a törölt fájlok visszaállítása. Amennyiben egy operációs rendszert is tartalmazó diszk főkönyvtárában visszaállítunk egy törölt fájlt, a továbbiakban nem lehet a diszkről rendszert tölteni. Ez floppylemezekre és harddiszkekre is egyaránt igaz. Sajnos az így elrontott lemezeken a SYS parancs sem segít, és az esetek többségében a Norton Disk Doctor is hatástalan.

A jelenség megértéséhez tekintsük át röviden a DOS könyvtárszerkezetét és annak kezelési szabályait. Minden DOS lemezegység tartalmaz egy ún. főkönyvtárat, amelyben a lemez kapacitásától függően korlátozott számú fájl-, illetve alkönyvtárbejegyzés helyezhető el. Minden bejegyzés 32 bájtot foglal el, és tartalmazza a fájl nevét, kiterjesztését, attribútumát, létrehozásának dátumát és idejét, méretét, valamint kezdő alapegységének sorszámát. Az alkönyvtárak speciális fájlként értelmezhetők, azok mérete nem korlátozott, így tetszőleges számú bejegyzést tartalmazhatnak. A DOS egy könyvtár tartalmát csak addig értelmezi, míg meg nem találja az első olyan bejegyzést, amelyben a név mező első karaktere 00h. Természetesen a bejegyzések törölhetők, azonban ez nem fizikai felülírás, hanem csupán a név mező első karakterének lecserélése E5h karakterrel. Ez azt jelenti, hogy a törlés megszüntethető, azaz a fájl (alkönyvtár) visszaállítható azáltal, hogy nevének első karakterét a név szabályainak megfelelő karakterre cseréljük — azonban figyelembe kell venni azt a tényt is, hogy a törölt fájl által elfoglalt lemezterület felülírásra kerülhetett a törlés és a visszaállítás közötti időben. Ilyenkor az automatikus visszaállítás nem lehetséges, a manuális visszaállítás pedig keresztláncolási hibához vezet.

Szabálysértés

A könyvtárbejegyzések sorrendjére nincs semmilyen előírás, egy kivétellel, ez pedig az operációs rendszert tartalmazó lemezek főkönyvtára. Mivel a rendszer betöltését végző ún. bootszektor mindössze 512 bájt hosszú programot tartalmazhat, így nincs benne elegendő hely olyan program számára, amely képes lenne a főkönyvtárban kikeresni az operációs rendszer fájljainak bejegyzéseit, és azokat beolvasva a rendszer töltését elvégezni. Ezért az ilyen lemezeken a főkönyvtár első két bejegyzésének kötelezően a két rendszerfájl (pl. IO.SYS, MSDOS.SYS) adatait kell tartalmaznia. Ellenkező esetben a rendszer betöltése lehetetlen. Általában ezek a fájlok rejtett, csak olvasható és rendszer- (HSR) attribútummal ellátott állományok — annak érdekében, hogy az egyéb programok számára jelezzék, ők nem mozdíthatók, nem nevezhetők át, nem törölhetők, és könyvtárbeli helyzetük sem változtatható.

Ezt a szabályt nem tartotta be a Disktool szerzője. A program törölt fájlok visszaállítására szolgáló menüpontja miután beolvasta a főkönyvtár tartalmát, automatikusan névsorba rendezi azt úgy, hogy előreveszi az alkönyvtárbejegyzéseket, majd ezek után helyezi el a fájlbejegyzéseket, így — hibásan — áthelyezi az operációs rendszer fájljait is a rendezésnek megfelelően. Ilyen esetben a SYS parancs nem hatásos, mivel az csak abban az esetben engedi meg az operációs rendszer elemeinek felmásolását az adott lemezegységre, ha annak főkönyvtárában az első két bejegyzés szabad.

Inkább mással

A fentiek alapján egyetlen lehetőség marad az elrontott, operációs rendszert is tartalmazó diszkek helyreállítására. Meghatározni a rendszer fájljainak nevét, kikeresni azokat a főkönyvtár bejegyzései közül, és felcserélni őket az első két bejegyzéssel. Így a diszk újra használható lesz. Erre szolgál a lemezmellékleten található FIXUPDT.EXE nevű program, amely a paramétereként megadott meghajtón ellenőrzi a főkönyvtár tartalmát, és ha szükséges, a megfelelő korrekciót végrehajtja. A program egyaránt használható harddiszkek és floppyk esetén is.

A Disktool tesztelésének eredményei alapján úgy véljük, hogy a program nem nyújt olyan szolgáltatást, amely kiemelné a hasonló tudású más programok közül, azaz használatát indokolttá vagy szükségessé tenné. Helyette inkább a PC Shell, illetve a Norton Utility egyes programjait javasoljuk, amelyek lényegesen megbízhatóbban működnek.

Farmosi István

A mérnökök ízlése szerint

Szabad egy Tangóra?

Az amerikai EDN magazin egyik ez évi számában érdekes összehasonlításra bukkantunk. A csaknem 300 tervezőmérnök véleményére alapozott teszt eredményei alapján úgy gondoljuk, hogy nekünk is érdemes nagyító alá venni a Magyarországon még kevéssé ismert Tango nyomtatottáramkör-tervező rendszert. A kaliforniai Accel Technologies cég terméke ugyanis az egyik legnépszerűbb elektronikai CAD rendszer az Egyesült Államokban csakúgy, mint Európa nyugati felében. A múlt év végétől a már itthon is beszerezhető programcsalád legfrissebb verzióit „táncoltatjuk meg" — két fordulóban.

A titokzatosan csengő Tango elnevezés PC-kompatibilis számítógépekre készített elektronikai tervező programcsaládot takar. A szoftver első verziója 1988-ban látta meg a napvilágot, ez a dátum két szempontból is érdekes: egyrészt magyarázza a Tango viszonylagos ismeretlenségét az „öregebb" vetélytársakkal szemben, másrészt magában hordozza a termék korszerűségét — hiszen már az első verziók is modernnek nevezhető hardver- és szoftverkörnyezetben születtek.

Hogy elemében érezze magát...

A Tango rendszer elemei — néhány vetélytárssal ellentétben — csakis PC-ken használhatók, ott viszont igazán elemükben érzik magukat. Az 1993 februárjában bemutatott Tango PRO for Windows egy valódi 32 bites, Windows 3.1 operációs rendszer alatt futó program. A szoftver 386/387 vagy 486-os mikroprocesszort igényel, és ténylegesen kihasználja azok és a Windows minden vonzó lehetőségét. A Windows és a Tango fejlesztői közötti folyamatos és szoros együttműködésnek köszönhetően várható, hogy a Windows NT hivatalos megjelenését követő rövid időn belül piacra kerül a Tango új, Windows NT alatt futó változata is.

A korábbi kiadású, DOS alatt futó, 16 bites Tango-modellek azonban szerényebb hardverrel is megelégszenek. A 640 kbájt memórián felül a nagyobb feladatokhoz EMS-bővítés használható. Ajánlott a VGA (vagy EGA) monitor — bár szükség esetén egy Hercules is megteszi. Számíthatunk széles körű nyomtatótámogatásra: a 9 és 24 tűs mátrixnyomtatóktól a tintasugaras printereken keresztül a lézernyomtatókig szinte mindennel dolgozhatunk. Természetesen az ismert plotterek is illeszthetők a Tangóhoz.

Termékválaszték

A nyomtatott áramkörök tervezésének minden lépését tartalmazó Tango PRO for Windows csúcsmodell mellett a hasonló képességű DOS-os verzió és számos egyszerűbb változat is beszerezhető a Tangóból.

A Tango-Schematic nevű termék a kapcsolásirajz-szerkesztő programot, valamint a 20 000-nél több elemből álló alkatrészkönyvtárat tartalmazza. A Tango-PCB a nyomtatottáramkör-tervező program, a Tango-PCB PLUS pedig ennek továbbfejlesztett változata. A Tango-Route segítségével — amely a PCB-vel vagy a PCB PLUS-szal is együttműködik — automatikusan huzalozhatunk legfeljebb 4 rétegű paneleket. A Route PLUS ugyanezt 8 rétegen, bonyolultabb tervek esetében is elvégzi. A Tango-Route PRO a legkomolyabb igényeket is kielégítő, 100%-os automatikus huzalozó. A programozható logikai eszközökkel történő áramkörtervezést és -szimulációt a Tango-PLD modul teszi lehetővé. Természetesen az itt felsorolt termékek egyedileg és többféle csoportosításokban is hozzáférhetők.

Összehasonlítva másokkal

A már említett EDN-teszt két területen mérte össze a piac által legjobbaknak tartott nyomtatottáramkör-tervező rendszereket. Az egyik kategóriában a kapcsolásirajz-szerkesztők, a másikban a tényleges paneltervező programok vetélkedtek egymással. Az összehasonlítás végeredménye egy szám lett — 1 és 100 között —, amely a fontosabb szolgáltatások és tulajdonságok súlyozott átlagértékét tükrözi. Mindkét kategóriában — a felhasználók véleményei alapján — a legfontosabb szempontnak a termék megbízhatóságát, a használat egyszerűségét és a kivitelezés minőségét tartották. Befolyásolta a végeredményt, hogy a termék ára mellett az egyes programok dokumentációi mennyire jók, milyenek a rendelkezésre álló technikai információk és szervizlehetőségek. A kategóriánként különböző tulajdonságok olyan technikai paraméterek, amelyek meghatározzák a termék alkalmazhatóságát.

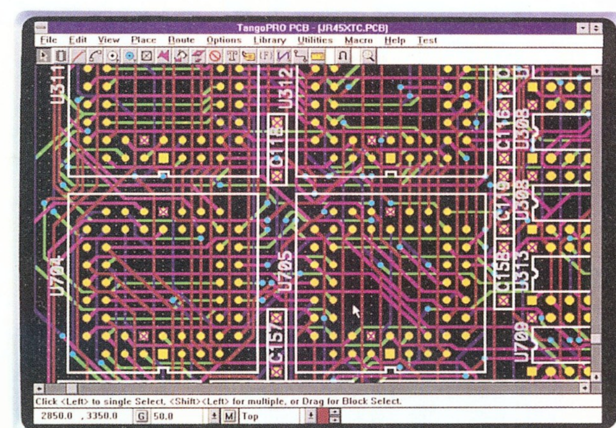
A kapcsolásirajz-szerkesztő programok 7 tagú mezőnyében további lényeges szempont volt a rendelkezésre álló alkatrészkönyvtárak mérete és minősége, valamint a szabványos illeszkedési felületek megléte. A szorosnak mondható mezőnyben a Tango az előkelő harmadik helyezést érte el, maga mögé utasítva ezzel több, a szakmában nagy tiszteletnek örvendő nevet.

A paneltervező programok versenyére 5 indulót neveztek, és a viadal igen szorosnak bizonyult, hiszen az összesített pontszámok legmagasabbika 72, a legalacsonyabb pedig 65 volt; cikkünk ihletője az utána következő második helyezettet 5 ponttal előzte meg.

Ezek az eredmények — bár túlnyomórészt egyesült államokbeli mérnökök véleményét tükrözik — jogosan keltik fel a magyar szakemberek figyelmét is. A hazai elektronikai fejlesztések visszafogottsága ellenére is jó vásárnak bizonyul a Tango, hiszen rendkívül kedvező ár/teljesítmény arányokat mutat.

Megbízható könyvtárakban

Ezek után lássuk, hogy mivel is nyerte el az amerikai mérnökök bizalmát a Tango! Az ipari alkalmazások megítélésében egyre nagyobb hangsúlyt kap a megbízhatóság. Bár a Tango alkatrészkönyvtárai nem a legnagyobbak a mezőnyben, kiemelkedő pontosságuk és hibamentességük miatt mégis sok tervező esküszik rájuk. Mind a kapcsolásirajz-szimbólum, mind pedig a nyomtatásialkatrész-könyvtárakat kezelő programrészlet hasznos tulajdonsága az elemek azonnali grafikus megjelenítése keresés közben. Így lényegesen gyorsabban megtalálhatók a keresett szimbólumok akkor is, ha azok neve nem, vagy nem pontosan ismert.



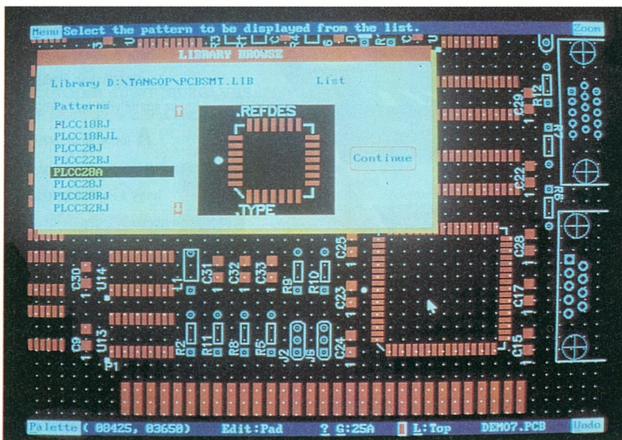
Egy produktív programinterfész

A kezelhetőség és a gyors megtanulhatóság sem kevésbé elhanyagolható szempont. A kezelői felület kialakításához tanulmányozták a mérnökök munkaszokásait, a tervezési folyamat lépéseit, a más programokban alkalmazott módszereket, és figyelembe vették a felhasználók javaslatait. A gondos tervezés eredménye az Accel Productivity Interface (API), azaz produktív programinterfész, amely billentyűzettel és egérrel egyaránt kényelmesen kezelhető.

Minden Tango program kezelői felületét így alakították ki, tehát a használatukat megtanulni — arra ráérezni — csak egyszer kell. Így a felhasználó mentesül az egyébként jó dokumentáció vagy a helyzetérzékeny help folyamatos használatától. Ennek a jelentőségét az érzékeli igazán, aki dolgozott már más tervezőprogrammal, amely bár sokat tudott, bonyolult kezelhetősége sokakat mégis inkább elriasztott.

Az API legördülő menük, dialógusablakok, közvetlenül elérhető almenük, automatikus rácspontra helyezés, beépített súgó, és még tucatnyi hasznos tulajdonság segíti a tervező munkáját.

Valamennyi program képernyőképe azonos felépítésű. A négy sarokban elhelyezett négy gyorsbillentyűvel a leggyakoribb parancsok hajthatók végre, a felső sorban pedig mindig az aktuális paranccsal kapcsolatos információk olvashatók. A képernyő alsó sorát mezőkre osztották, ahol a program pillanatnyi állapotát jelző információk (kurzorpozíció, lapméret, állománynév...) mellett egy szokatlan, de rendkívül jól használható menü- és almenüelérési lehetőség is szerepel.



Nem elszigetelt jelenség

A korszerű tervezőrendszerek világában az illeszkedési felületekkel nem rendelkező, elszigetelt programok nem lehetnek hosszú távon életképesek. A Tango azonban képes különféle formátumok ki- és bevitelére.

A kapcsolási és nyomtatási rajzok Postscript formátumban is előállíthatók. A nyomtatott áramköri lap kimenthető DX-formátumban, amely lehetővé teszi, hogy az áramkört magában foglaló készülékház terveibe közvetlenül beilleszthető a panel — mechanikai alkotóelemként.

Bár a PCB modulok szervesen illeszkednek a Schematic kapcsolásirajz-szerkesztő modulhoz, velük az OrCAD kapcsolási rajzok is feldolgozhatók. Az amerikai eladási visszajelzések alapján az elterjedt tervezőrendszer-kombináció az OrCAD kapcsolásirajz-szerkesztő és a Tango PCB vagy PCB PLUS.

A Tango legfontosabb általános tulajdonságainak bemutatása után az Alaplap következő számában a kapcsolásirajz-szerkesztő és a huzalozó modulokról informálódhatunk részletesebben, és raadásként egy igazi meglepetést is tartogatunk a Tango iránt érdeklődő olvasók számára.

Lóth Tamás

Legnagyobb „áruház" — legnagyobb választék



Magyar szakemberek is szép számban látogatják a világ legnagyobb számítástechnikai rendezvényét. Utána azonban szidják. Nem vitatjuk, hogy a hatalmasra nőtt CeBIT meglehetősen strapás. Többnyire valóban elég kényelmetlen dolog oda- és visszautazni, akármilyen szállást találni, a bejárhatatlan méretű kiállítási területen a tömegben araszolni, a bennünket érdeklő standokhoz hozzáférni, a kompetens személyeket megtalálni és velük időpontot egyeztetni...
Mindezt mégis évről évre egyre többen vállalják. Vajon miért?
Mi például egész éven át profitálunk a CeBIT-en összegyűjtött anyagból és az ott felvett szakmai kapcsolatokból. Mindenki számára más okból és más módon lehet üzletileg érdekes vagy szakmailag megragadó, ötletadó, gondolatébresztő.
Mostani összeállításunkban azt mazsoláztuk ki a CeBIT-ből, ami tendenciát, iránytartást vagy irányváltást jelezhet — feltételezésünk szerint az Alaplap olvasóinak többségét érdeklő témákban.

A mágnes

A kedvezőtlen gazdasági helyzet alapján a vásároknak is megjósolt visszaesés a CeBIT-et egyelőre elkerülte. Idén is mágnesként vonzotta a számítástechnikai szakmát. Hiába jelentették be távolmaradásukat olyan neves cégek, mint a Borland vagy a Novell, végül is ők maradtak a kivételek. (Vajon az 1994-es CeBIT-et is kihagyják?) A résztvevők száma (5 604 kiállító cég, köztük 2 136 külföldi) meghaladta a tavalyit. A nettó kiállítási terület szintén nőtt, 306 ezer négyzetméterről 313 ezerre. Előzetesen a látogatók számának várható visszaesését is jósolgatták. A tények erre is rácáfoltak: az 1992-es CeBIT-en regisztrált 648 900 látogatóval szemben idén 660 ezren voltak, köztük 104 ezren külföldről. Ennyi megtévedt ember? Vagy talán mégis van valami haszna a vásárlátogatásnak?



Sok újságíró fanyalogva járta a vásárt, hogy „nincsenek igazi szenzációk". Való igaz: a szaksajtó számára is kellenek olyan hírek, amikor „a postás harapja meg a kutyát", mert fordítva a dolog túlságosan hétköznapi, banális. Lassanként azonban le kell szoknunk a bombasztikus vásári tudósításokról, a „merő véletlenségből" éppen a vásáron megkötött nagy üzletekről és az ugyancsak véletlenül a vásáron felbukkant „vadonatúj" termékekről. Az üzletkötések vásári időzítése régebben is csak görögtűz volt, az igazi technikai újdonságoknak a vásárokig történő „jegelése" pedig napjainkban szinte lehetetlen, mert a piac kikényszeríti a mielőbbi bejelentést, de még a kiszivárgó hírek is egyidejűleg és egyszerre (real time és on-line) jutnak az egész világ tudomására, nem várva meg semmiféle vásári ceremóniát.

A CeBIT-en a kiállítóknak és a látogatóknak tehát el kell viselniük a nagyság átkát, a zsúfoltsággal járó megpróbáltatásokat, mert ez a koncentrált és sokoldalú bemutató egyelőre a leghatékonyabb módja a kínálatról való tájékozódásnak, és a számítástechnika területén más európai vásárok — amint mellékelt ábránk is mutatja — ezt a teljességet meg sem közelítik.

A mágneslemez

Minden szentnek maga felé hajlik a keze. Mi is megkülönböztetett figyelemmel vizsgáltuk az Alaplapot közvetlenül érintő tendenciákat. Így például lemezmellékletünk perspektíváit. Azt kellett látnunk, hogy hosszú ideig tartó bizonytalankodás és kettősség után a 3,5"-os lemezméret most már kezd fölénybe kerülni. Ennek egyik fő oka nyilvánvalóan a laptop gépek elterjedése.

Logikus, hogy miután a hordozható modellek 3,5"-os meghajtóval vannak felszerelve, az asztali gépekben is ezt kell uralkodóvá tenni. Csak egyetlen apró jelzés: az idei

CeBIT-en a cégektől már minden demóprogramot 3,5"-os floppyn kaptunk meg.

Az Alaplap áprilisi számában elhelyezett közvéleménykutató kérdőív segítségével ilyen szempontból is igyekszünk felmérni a hazai helyzetet, esetleges változtatásokat azonban csak a tényleges hardverellátottság ismeretében tervezünk. A tendencia persze nyilvánvaló, és ha manapság valaki új gépet szerez be, kérjük, ne spórolja ki belőle a 3,5"-os meghajtót.

Árvíztűrő tükörfúrógép©

A fenti kis „terminus technicus" szerkesztőségünk évek óta jól bevált ékezettesztelője. Új szövegszerkesztők vagy betűkészletek (fontok) ékezethelyességének vizsgáztatására elég csupán ezt a két szót beírni, kis- és nagybetűs változatban, és minden kiderül. Így tettünk Hannoverben is a magukat soknyelvűként propagáló programokkal.

Annak a fejleménynek, hogy lassanként Nyugat-Európában és Amerikában is össz-európai szinten gondolkodnak, van egy számunkra igen kedvező következménye. A fejlesztők végre kezdik belátni, hogy a számítógépes programoknak a kisebb európai népek fránya ékezetes betűit is „ab start" korrektül kell kezelniük. Örvendetesen gyarapodott például



azon szövegszerkesztők száma, amelyekben át lehet váltani más európai nyelvekre, sőt ablaktechnikával az éppen használt nyelv billentyűzetének rajza is állandóan a képernyőn, azaz szem előtt tartható. Van olyan program, amely csak indításkor mutatja meg a magyar billentyűzetkiosztást, utána fejből kellene tudni, hogy mi hol van, de ha kinyomtatjuk a rajzot, azt puskának használhatjuk.

A szoftverek magyar nyelvi szempontból nem egyformán kidolgozottak. Ilyesmiben üt vissza az is, hogy annyiféle kódkiosztást és billentyűzetkiosztást használunk. A külföldi fejlesztőnek melyikhez kellett volna igazodnia, amíg nem volt egységes magyar szabvány? Minden kezdet nehéz, és semmi nem megy egyszerre. Az Accent nevű programban például a hosszú magyar „ő" helyén még a fordított ékezetes „y" jön be, a magyar billentyűzetrajz pedig hiányzik, de „már dolgoznak a hiba elhárításán".

A Windows grafikus felületnek mindenképpen javára kell írni, hogy „ablakot" nyitott az egész világra, a szövegszerkesztésbe integrálhatóvá tette még a nem ábécé rendszerű írással rögzített egzotikus nyelvek jelkészletének használatát is. Erre jó példa a Gamma UniVerse for Windows itt bemutatott képernyőfotója.

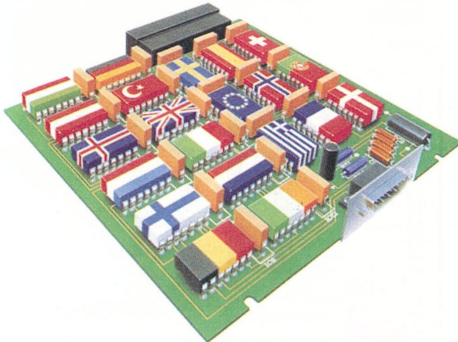
Ha már a szövegszerkesztőkről esik szó, érdekes volt megfigyelni a WordPerfect stratégiájának megváltozását. E klasszikus szövegszerkesztőnek egy darabig kiadványszerkesztő (DTP) karriert is szántak, hogy felvegye a versenyt a PageMakerrel és a Venturával. Most úgy tűnik, hogy ezt a reménytelen harcot feladta, és 6.0-ás új változatával inkább azt célozza meg, hogy továbbra is ő legyen a legsikeresebb szövegszerkesztő DOS és Windows felületen egyaránt, hiszen piaci részaránya az Egyesült Államokban jelenleg 80%! Egyidejűleg határozott lépéseket tett más területek meghódítására, mint a táblázatkezelés (PlanPerfect), az adatbáziskezelés (DataPerfect), a prezentációs grafika (DrawPerfect, WordPerfect Presentations) vagy a komplex irodai szoftverek világa (WordPerfect Office).

Az Édentől keletre

Folytatódott az először tavaly megrendezett „Software in Europe" akció, melynek keretében több mint 20 országból mintegy 250 programot láthattunk. Annak ellenére, hogy kissé erőltetettnek tűnik a kiállításon belül ilyenfajta elkülönítés, a 4 győztes program mindenképpen figyelmet érdemel. A MarketForce belga marketingszoftver és a FOSS angol termelésirányítási rendszer mellett eredetiségével emelkedik ki a finn IDEGEN++, amely ötletek generálásával, kritikai szűrésével fejleszti kreatív gondolkodásunkat. A bolgárok Alpha Series díjnyertes komplex szoftverével egyebek között 30 európai nyelven tudunk szöveget szerkeszteni, és DTP-re, prezentációs célokra is kitűnő.

A tavalyihoz képest kevesebb magyar kiállító volt az idei CeBIT-en. A világszerte (el)ismert Recognita és ArchiCAD azonban nem maradt távol. A Recognita most bemutatott új verziójában a felismerés pontosságát jelentősen növelték, és többféle szkennert illesztettek a programhoz. E szoftver nemzetközi jelentőségét jól jelzi, hogy a Microsoft standján, a kiállító partnercégek között immár a Recognita is helyet kapott.

A másik hazai csúcstermék, az ArchiCAD fejlesztői nem rukkoltak ki új verzióval — pedig sokan várják már a windows-os változatot. Ők „csak" megtervezték ArchiCAD-del a hannoveri vásárterület mindenki által megcsodált,



légiesen kecses, idén átadott pavilonját. Hoztak azonban más újdonságot: a TopCAD új verziója mellett bemutatták Next gépre a Merlin elnevezésű kétdimenziós, általános célú CAD rendszert. A CAD világánál maradva, a riválisnak számító, ugyancsak magyar fejlesztésű ArchiTECH.PC építészeti tervezőrendszer windows-os premierjének lehettünk tanúi.

Nem számított „első bálozónak" a KFKI Számítógéphálózatok Kft. A tavaly már bemutatott (és azóta külföldön is terjedő), saját fejlesztésű ENMC monitorozó programjuk mellett egy olcsó, amerikai hálózatmenedzselő szoftvert (SNMPc) mutattak be, amelyet kiokosítottak a meglehetősen heterogén magyar hálózati viszonyokra.

A CeBIT-en saját költségből résztvevő magyar kiállítók mellett idén a „Business with Eastern Europe" meghívás adott rendkívüli alkalmat néhány bemutatkozásra. A Neumann János Számítógéptudományi Társaság a multimédia eszközeivel illusztrálta a hazai számítástechnika történetét, a kiemelkedő alkotásokat és a társaság tevékenységét. A magyar vállalatok és vállalkozók iránt érdeklődők a Magyar Gazdasági Kamara országos, és a Magyar Vállalkozásfejlesztési Alapítvány megyei adatbázisából informálódhattak. Ugyancsak ebben a blokkban állította ki a Budapesti Műszaki Egyetem a Multivox szöveg-beszéd átalakító szoftvert, amely 8 nyelven fordít szövegről hangra, és komolyan érdeklődött iránta több multimédiával foglalkozó cégen kívül a SoundBlaster is.

Messze van még Amerika?

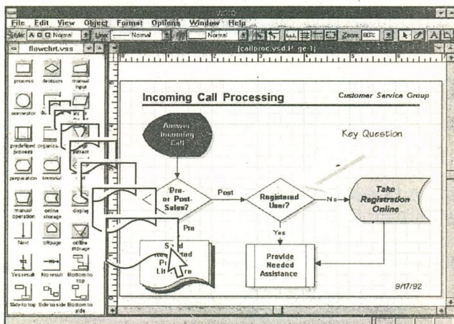
A felhasználói szoftvereket nézegetve szembeötlő volt, hogy a legtöbb program grafikus felületet használ. A Windows térnyerése úgy látszik megállíthatatlanul folytatódik, gőzerővel írják át a programokat Windows alá. A hatalmas apparátussal felvonult Microsoft óriási standjait non-stop bemutatók övezték, a kör közepén pedig a partnercégek állították ki fejlesztéseiket. A érdeklődők szüntelen áradata miatt a bemutatókhoz alig lehetett közelférkőzni.

Más cégek standjait is elárasztották a Windows applikációk, sőt jónéhány helyen ott hivalkodott a Windows NT is, amelyről még nehéz tárgyilagos véleményt alkotni. Számunkra elég zavaró, hogy miközben a szakirodalom minden jót ír, eddig még nem találkoztunk olyan hazai számítástechnikussal, aki a Windows NT rendszert fél óránál hosszabb ideig tudta volna futtatni...

Érdekes a printer/plotter front alakulása. Egyre több a fénykép minőséghez közelítő színes nyomtató, egyre több a 600 és 1200 dpi felbontású nyomtató, és úgy látszik, hogy sikeres áttörést hajtott végre a tintasugaras technológia, nemcsak a mátrixprinterek helyére pályázva, hanem a plotterek kategóriájában is. Az alapanyag pedig ezáltal kezd egyre változatosabbá válni: pausz, műnyomópapír, fólia, műanyag... sőt textil.

Miközben sok számítástechnikai területen Magyarország lemaradása nem látszik vészesen nagynak, azt még elég nehéz elképzelni, hogy a multimédia alkalmazásában hamar követni tudjuk a CeBIT-en látottakat. Nagyon sok program már eleve CD-ROM-on kerül forgalomba, amihez külön berendezés szükséges. Emellett hangkártyák, hangszórók, adapterek, videoillesztések és más tartozékok szinte korlátlanul bővíthető arzenálja vonul fel az új irányzat támogatására.

Hogy önkényesen kiragadjunk valamit: az amerikai DeLorme Mapping cég világatlaszán (a rajzprogramokból is-



mert nagyítókeretes fókuszolással) az egész féltekét ábrázoló képernyőtől néhány egérmozdulattal eljutottunk Budapestre, a Keleti pályaudvar környékének részletes rajzáig. A szoftver egyúttal a műemlékeket, szállodákat, kulturális létesítményeket is jelzi. Ez a szoftver önállóan is használható, de még hatásosabb, ha beillesztik egy komplex multimédia rendszerbe, ahol a látnivalók videóval azonnal megeleveníthetők, a hozzá kapcsolódó információk az adatbázisból behívhatók, az adatokkal interaktív műveletek végezhetők... Ahol mindaz megtehető, amit csak képzeletünk és a rendelkezésünkre álló eszközpark megenged. Előbb-utóbb nálunk is! .

Pentium

Nem meglepő, hogy a DEC 150 MHz-es Alpha-PC-je mellett a CeBIT szenzációjává az Intel 486-os processzorainak „trónörököse" vált, amely 586-osnak indult, P5-ösként futott a célegyenesbe, és végül Pentium néven jelent meg — egyelőre még „lefátyolozva". Az Intel standjára elvándorló tömeg üveg mögött (!) nézhette meg a Pentiumot, s csak néhány kiváltságos kerülhetett érintésközelbe a 60 és 66 MHz-es változatban készülő, 3,1 millió tranzisztort tartalmazó lapkával.

Az i486-os processzornál két és félszer több tranzisztorból álló Pentiumról naponta 3 alkalommal hangzott el előadás a konferenciaközpont mindig zsúfolásig megtelt nagytermében. Nem a technikai paraméterekkel bombázták az érdeklődőket, mert azok már előbb is hozzáférhetőek voltak, s a Mark & Technik 310 oldalas szakkönyvét („A Pentium processzor") a helyszínen meg lehetett venni. Az előadók inkább azt próbálták érzékeltetni, hogy különféle hardverplatformokon és eltérő operációs rendszerek alatt (DOS, Windows, OS/2, Unix, SCO Unix, Windows NT stb.) az egyes alkalmazásoknál mekkora sebességnövekedés érhető el. A 486-osokhoz képest átlagban négyszeres gyorsaságot jól érzékeltette az a bemutató, ahol egy i486DX-33, egy i486DX2-66 és egy Pentium processzoros HP gépen egyszerre indították el ugyanazt az alkalmazást.

A Pentium iránt óriási az érdeklődés a hardvergyártók részéről. A Compaq például mindent megpróbált, hogy a CeBIT-en bemutathasson egy Pentium processzoros gépet, de az Inteltől végül is nem kapta meg a hozzájárulást, ezért a többi hardveres céghez hasonlóan kénytelen volt beérni a „Pentium is coming soon" felirattal.

Faklen Pál—Sziebig Andrea

Ellenszer a lemezmellékleten

SWB — a májusi gyilkos

Ezt a bootvírust Magyarországon 1992 decemberében lokalizáltuk első ízben az egyik budapesti egyetem számítástechnikai tanszékén. Szerencsére a tanszék dolgozói idejében észlelték a fertőzést.

Ez év első két hónapjában több budapesti és vidéki felhasználó jelezte az SWB felbukkanását. Tömeges elterjedése ez év áprilisára, májusára várható, romboló hatása május 21-ére van beprogramozva. Nem is akármilyen pusztítást végez, gonoszabbat, mint a közismert Stone vírus, amely „csak" használni nem engedi a gépet, de végleges adatvesztést nem okoz.

Az SWB minden év május 21-én leformázza azt a lemezmeghajtót, amelyikről betöltődött. Működése, fertőzési mechanizmusa megegyezik a többi bootvírus által használt technikával. Azt használja ki, hogy a gépek BIOS-a — eltekintve az újabb gépektől, amelyekben a rendszertöltés sorrendje beállítható — először mindig az A: floppyegységről kísérli meg az operációs rendszer betöltését, és csak abban az esetben fordul a gép merevlemezéhez, ha ez onnan nem sikerül. A fentiekből következően a gépet meg lehet fertőzni azáltal, hogy fertőzött floppy-lemez van a meghajtóban, és a gép arról kísérli meg az operációs rendszert betölteni. Természetesen az sem jelent akadályt a vírus számára, ha a lemezen nincs operációs rendszer, bootszektora ugyanis van, s az lecserélhető a víruskódra. Mivel a BIOS nem ellenőrzi a bootszektor tartalmát, azt minden további nélkül betölti, és a benne lévő programot el is indítja. Fertőzött lemez esetén ez a program maga az SWB vírus, amelynek első ténykedése, hogy beépül a gép memóriájának végébe, az aktív memória méretét 2 kilobájttal csökkenti, majd ellenőrzi, hogy a merevlemez fertőzött-e. Ha az még tiszta, természetesen azonnal megfertőzi, ezáltal biztosítva, hogy a következő rendszertöltéskor, a merevlemezről is aktivizálódhasson.

A vírus ellenőrzése alá vonja a gép lemezműveleteit, így a későbbiekben az A: floppyegységen elegendő egy DIR parancsot kiadni, és az ott lévő lemez azonnal megfertőződik. Szaporodását, terjedését így biztosítja. Romboló hatása drasztikus. Aktivizálódásakor a lemezegység első sávjától befelé haladva annak szektorait felülírja a memória egyik területének tartalmával (5000h:5000h). Ezáltal a lemezen lévő adatok teljesen megsemmisülnek, a helyreállítás lehetetlenné válik. Floppylemezeken azok mindkét felületét tönkreteszi, míg merevlemezek esetén „csak" az első 5 felület első 17 szektorát írja felül. Sajnos azonban ez is végzetes. Az így tönkretett merevlemezek csak ismételt particionálás és formázás után használhatók.

A vírus a Novell fájlszervereket is képes tönkretenni, és azok a felhasználók sem lehetnek biztonságban, akik nem DOS-környezetben üzemeltetik a PC-t, hiszen amikor a vírus egy floppyról aktivizálódik, még nincs a gépben operációs rendszer, maga a vírus sem tudja, de őt nem is érdekli, hogy milyen rendszer kerül majd betöltésre, Novell 2.xx, 3.xx, vagy netán OS2, esetleg Unix. Tapasztalataink szerint a Novell 2.xx szervereket a fertőzés ténye is azonnal tönkreteszi, míg a vírus a 3.xx Novell-verzió mellett vidáman megél. A Unix rendszereket a fertőzéssel egyidejűleg használhatatlanná teszi, azáltal, hogy lefoglalja a memória végét.

A mágneslemez mellékleten közreadott antivírus program alkalmas mind a floppyk, mind a merevlemezek ellenőrzésére és megtisztítására is. Paraméterként azon meghajtók betűjeleit kell megadni, amelyeket ellenőrizni kívánunk. A program — ha fertőzött lemezt talál — a helyreállítást is automatikusan elvégzi.

Farmosi István

The World's Best Selling UNIX Clone

Coherent 4.0

Teljes 32 bites változat 15 000 forint + áfa
Coherent 3.2 (286 processzorra) 12 000 forint + áfa

Egy teljes UNIX operációs és fejlesztőrendszer vár Önre!

BECO Kft.

1132 Budapest, Visegrádi u. 62. Telefon: 270-3299
Telefon/Telefax: 149-8580

DUAL NOTEBOOK

A választék:

386 SXL 25, 80 MB HDD	132.000.-
386 SXL 25, 80 MB HDD, Color	229.000.-
486 DX 33, 120 MB HDD	238.000.-
486 DX2 50, 120 MB HDD	259.000.-
486 DX2 66, 120 MB HDD	279.000.-
486 DX 33, 120 MB HDD, Color	313.000.-
486 DX2 66, 120 MB HDD, Color	334.000.-

Pocket FAXModem 9600 bps+MNP5 most csak 17.700 Ft!


Hoktrade Kft. 1012 Attila út 93. Tel: 202-4166, Fax: 175-0446

PRINTERSYS

Számítástechnikai és Szolgáltató Bt.
1043 Budapest, Aradi utca 15. Telefon: 169-2419

EPSON típusú lézernyomtatók, mátrixnyomtatók javítása, karbantartása, értékesítése



Számítógépek installálása, javítása, bővítése

Opciók Kellékek Festékkazetták

GARANCIA, MEGBÍZHATÓSÁG, BIZTONSÁG

PRINTERSYS

Számítástechnikai és Szolgáltató Bt.

EPSON Mintaboltja:

7623 Pécs, Munkácsy M. utca 9. Telefax: (72)36-226

INFORMÁCIÓKÉRÉS: A0554 ▼

Faxszenzációk

Borítékolja a faxot

A SEALFAX előnye, hogy ugyanolyan bizalmasan kezelhetők általa a vett faxok, mintha ajánlott levelet kapott volna a címzett. A beérkezett faxok szétválogatásával, másolásával sem kell az időt tölteni. A vett üzeneteket a berendezéshez tartozó finom felbontású lézerfax A/4-es formátumban kinyomtatja, majd a berendezés félbehajtja és egy környezetbarát fóliaborítékba becsomagolja. A fólián csak az első lap fejléce látszik át. Ezen látható a címzett neve. A további lapokat már takarja az első lap, így azokat illetéktelen személy nem olvashatja el.

SEALFAX

Nagy teljesítményű hálózati faxszerver

A FaxDirector egy „tárold és küldd tovább" telefaxrendszer X.25 és SNA hálózatokhoz továbbküldés, postaláda és információ-visszakeresési módokkal. A jellemzői röviden: E-MAIL integrálás beépített X.400 gateway-n keresztül, 3270 host kapcsolat és LAN faxszerver lehetőség DOS, Windows és OS/2 munkaállomásokhoz. Automatikus konverzió Group 4-re, kapcsolás/útvonalkeresés X.25, SNA és TCP/IP hálózatokon. FaxDirector csomópontok összefogása WAN hálózatba. Moduláris felépítés, egy modul maximum 16 telefaxkártyát képes kiszolgálni. Modulonként négy nagysebességű lézernyomtató és egy nagysebességű szkenner csatlakoztatható.

FaxDirector

Kizárólagos magyarországi forgalmazója:  **SECOTEL Kft.**

1116 Budapest, Fehérvári út 120.
Telefon: 161-0475 Telefax: 117-7241 Telex: 22-1805

Ha lúd, legyen CD-ROM-on!

Univel — UnixWare

A szóról — UnixWare — minden valamirevaló novelles számítástechnikusnak az jut eszébe, hogy NetWare, és ennek még valami köze van a Unixhoz is. Ez pontosan így van, tudniillik a Novell és a Unix System Laboratories (USL) közös céget alapított Univel néven — azért, hogy a Unix és a NetWare előnyeit egyesítve új, hálózati Unix rendszert hozzon létre.

Az alapötlet az, hogy a NetWare 386 v3.x változatú hálózati operációs rendszerek befogadják a Unix hálózatvezérlő részét NLM (NetWare Loadable Module) formában (a Unix System V Release 4.2 alapján). Ezután a NetWare a Macintosh fájlnevekhez hasonlóan a Unix fájlneveket is tudja kezelni (name space), ami köztudomásúlag különbözik a DOS fájlnevektől. Persze azért a DOS fájlrendszerről sem kell lemondania az újdonsült Unix-felhasználónak, mivel a NetWare azokat továbbra is kezeli.

Mi a helyzet a felhasználói felülettel, azaz hogyan jelenik meg a UnixWare a munkaállomáson? Pontosan mint egy X Window System, ami gyakorlatilag az MS Windows unixos megfelelője. A többlet csupán annyi, hogy a UnixWare X Window felületébe integrálták a NetWare fájlszerverek elérési lehetőségét is.

A UnixWare a saját szerverét Application Servernek hívja, és nem kevesebb, mint 120 Mbájt szabad winchesterkapacitást és minimum 12 Mbájt RAM-ot kíván. A UnixWare munkaállomásokon megjelenő része az ún. Personal Edition, amely megelégszik a munkaállomáson 80 Mbájt winchesterrel és 8 Mbájt RAM-mal (ha több, az nem baj). Emellett még illik, hogy a munkaállomás is minimum 386-os gép legyen (a 32 bites rendszer miatt) legalább 16 MHz-es órajellel.

Az installációs kézikönyv egy egész táblázatot közöl arról, hogy mennyi fér el a UnixWare-ből, ha csak 120 Mbájt winchesterünk van, ha 200 Mbájtunk van, esetleg még annál is több. (Valahol azt írtam, hogy az MS Windows winchesterzabáló operációs rendszer, hát akkor ez mi? Ráadásul a UnixWare nem a DOS-partíciót használja a winchesteren, hanem legalább két másik partíciót hoz létre magának, úgyhogy azon a gépen, ahol UnixWare-t használunk, felejtsük el a DOS-partíciót. Ha mégis DOS programokat akarunk futtatni, akkor két lehetőségünk van: vagy megtartunk egy kisebb DOS-partíciót a winchesteren (különösen, ha elég nagy az a winchester), vagy azt mondjuk a UnixWare-nek, hogy DOS programokat is akarunk rajta futtatni, de azokat valamelyik saját partíciójában lesz szíves tárolni.

A UnixWare természetesen tudja futtatni a DOS programokat, sőt úgy szállítják, hogy van benne egy kis DR DOS 6.0, de a kézikönyv szerint bármikor lecserélhető MS DOS 3.3-ra vagy 5.0-ra. A Univel egyébként azt ígéri, hogy még 1993 első félévében kijön a UnixWare MS Windows-emulációja, amellyel a Windows programok is futtathatók lesznek X Window System környezetben. Az X Windows maga tartalmaz egy-két MS Windowshoz hasonló segédprogramot is, kalkulátort, órát, szövegszerkesztőt stb. Ezenfelül az erőforrások is ikon formában érhetők el, a Norton Desktop for Windowshoz hasonlóan. A Macintosh operációs rendszeréhez hasonló viszont a „kuka", ahová az asztalról bedobhatjuk a már nem kellő ikonokat. Természetesen a kukából még nem tűnnek el véglegesen a beledobott dolgok, csak ha kiürítjük. Addig viszont lehet a kukában turkálni...

Ha már a nyílt rendszereknél tartunk, a Novell külön filozófiát dolgozott ki NICA (Novell Integrated Computing Architecture) néven. Ennek lényege, hogy különböző kategóriák különböző elterjedt rendszereit képes a NetWare és a UnixWare befogadni, illetve együttműködni velük. Ha a „kategóriák" szó az előbb egy kissé ködös volt, akkor most megvilágítjuk egy-két példán keresztül: a fájl- és nyomtatómegosztás (mint kategória) területén a NICA a DOS, Macintosh, OS/2, Unix természetes nyomtató- és fájlelérési rendszereket támogatja. A kommunikáció (mint kategória) területén pedig az IPX/SPX, SNA, TCP/IP, AppleTalk, OSI, T1, X.25 és az aszinkron rendszerekkel tud együttműködni. Az „Application Server" operációs rendszerek pedig az OS/2, Unix, MVS, VM, VMS, DOS és UnixWare rendszereket jelentik stb.

Ezeken az integrált rendszereken azért látszik, hogy jó pár mérnökévnyi munkát tartalmaznak, ezért sem az áruk, sem pedig a méretük nem mondható kicsinek. A winchester- és memória-helyigényen kívül ez látszik abból is, hogy mind az Application Servert, mind pedig a Personal Editiont CD-ROM-on, illetve QIC2-es streamerkazettán forgalmazza a Univel. Az alaprendszereken kívül léteznek még további segédprogramok (Personal Utilities), persze szintén CD-ROM-on, illetve különböző fejlesztői eszközök (Software Development Kit, Driver Development Kit és MTF Development Tools). A későbbiek során pedig a Univel kibocsátja a NetWare for UnixWare nevű terméket, ahol a NetWare összes szolgáltatása elérhető lesz UnixWare alól is.

Sík Zoltán

A mesterszakács és a hét törpe

Puhább közeg, keményebb feltételek

A feladatok sürgetőek. A legfontosabb szabály: nem szabad megállni, és nem lehet büntetlenül késlekedni. Amilyen fontos az új, hatékonyabb módszerek kiagyalása, ugyanolyan lényeges ezek gyors megvalósítása és kipróbálása, vagyis a tapasztalatgyűjtés is az újdonsült alkalmazási területeken.

A Snobolnak (legalább) három olyan fontos tulajdonságát, amelyek sok tekintetben más programnyelvek fölé emelik — legalábbis bizonyos típusú feladatok megoldásában —, bárki észrevehette. Az első talán az, hogy kevesebb figyelmet kell fordítani a járulékos problémák megoldására — ezek bizony gyakran megkeserítik programozók életét, főleg a kezdőkét. Nincs szükség a változók és a típus előzetes meghatározására és deklarálására (csupán a tömböknél). Gondmentes az interaktív programozás megszervezése: könnyű kitenni az adatokat a képernyőre, és bekérni a szükséges információkat. A másik kellemes tulajdonsága a Snobolnak, hogy rendkívül tömör programokat lehet benne készíteni. Néhány soros programmal lehet megoldani olyan problémákat, amelyek más programnyelvekben összehasonlíthatatlanul nagyobb nekikészülődést kívánnak: az eszköztár előzetes elkészítését, rutinok kifejlesztését, miegymást. Harmadikként említhetnénk a Snobol fortélyosságát, ami persze a fentebb említett másik két tulajdonságtól sem független: azért egyszerű a típusokkal bánni, mert a háttérben a Snobol maga elvégzi a típuskonverziók nagy részét, és azért tömörek a programok, mert erősek azok az eszközök, amelyeket felhasznál. A Snobol fortélyosságát mégis érdemes külön kiemelni, mert messzemenőkig kihasználja például azokat a lehetőségeket, amelyek a mintaillesztésben rejlenek.

A humán szféra meghódításáért

Napjaink egyik legfontosabb jellemzője, hogy a számítástechnika egyre inkább behatol a humán szférába is, azokra a területekre, ahol puhább a közeg és keményebbek a feltételek: a pszichológiába, a szociológiába, az orvostudományba, a nyelvészetbe és a nyelvtanításba, és hát persze a mesterséges intelligencia kutatásának ezerfelé terjeszkedő területére. Ezek a tudományágak nem kevésbé igényesek, mint a reál szféra — nem érik be az eddigi eszközökkel, hanem testhezálló módszerek kifejlesztésére kényszerítenek.

A mintaillesztés és a visszalépés alapvető fontosságát az intelligens feladatmegoldásban elsőként a mesterséges intelligencia kutatásának úttörői ismerték fel. Azóta ezek a módszerek a logikai programozásnak is legerősebb fegyvereivé váltak. „Leváltásukról" szó sem lehet, legfeljebb finomításukról, újabb módszerekkel való kombinálásukról.

A Snobol nyelv már régóta használja, mégpedig igen rafináltan ezeket a módszereket. Jelenleg a Snobol egyik legismertebb fejlesztője és szószólója, Ralph E. Griswold professzor munkatársaival együtt a grafikus lehetőségek irányába próbálja továbbfejleszteni ezeket a furfangos eszméket. Most kifejlesztés alatt álló nyelvüknek (amely azonban még távolról sem tekinthető kiforrottnak) az ICON nevet adták, és eddigi eredményeik alapján igen nagy reményeket fűznek a kutatási témához. A kutatás fellegvára az amerikai Arizona Egyetem, ahol több laboratórium szoros együttműködésben dolgozik hasonló jellegű témákon: a Gépi Nyelvészeti Laboratórium, a Pszicholingvisztikai Laboratórium, a Nyelvfeldolgozási Laboratórium és a Számítástudományi Laboratórium — a Kognitív Tudományok Munkaállomásának aktív közreműködésével.

Az Arizonai Egyetem nem áll egyedül a humán területekre is kiterjedő széles körű érdeklődésével. Hasonló tendenciát figyelhetünk meg szerte az Egyesült Államokban. Jellemzésül: egy frissen publikált részletes felmérés szerint egyedül az Egyesült Államokban 81 egyetemen folyik ilyen jellegű kutatás és továbbképzés. Kár volna lebecsülnünk természetesen Európa 45 egyetemi kutatóközpontját sem, amelyek között azonban sajnálatos módon nem szerepel a budapesti egyetem. (Magyar nevek viszont elég szép számmal vannak a külföldi kutatók között is.)

A fűszerek választéka

A mintaillesztés két legfontosabb követelménye, hogy viszonylag könnyen és gyorsan ki lehessen dolgozni elég frappánsan használható mintákat, és hogy a minták alkalmazása, újabb és újabb alternatívák keresése minél jobban automatizálva legyen.

A Snobol nyelv hatékony mintagyártó függvényekkel segíti a mintakészítés folyamatát. Hébe-hóba kipróbáltuk már ezek nagyobb részét, itt az ideje, hogy összegyűjtsük a teljes választékot. Lesznek köztük olyanok is, amelyeket később fogunk alaposabban kiismerni, a jó háziasszony gondosságával azonban ezeket is feltesszük a polcra.

1. Aktív változók

Vegyük sorra először azokat a változókat, amelyeknek értéke nem megadott argumentumoktól függ, hanem attól a helyzettől, amelyben alkalmazzuk őket. Kezdeti értéküket maga a Snobol rendszer állítja be, későbbi értékeik azonban egyedi viselkedési módjuktól függenek.

Hét mintaelem tartozik ebbe a csoportba, mindegyiknek megvan a maga sajátos viselkedési módja — akár a hét törpének, ha nevük kevésbé fejezi is ki jellegzetességüket.

Íme a hét törpe: a REM, az elmaradók felkarolója; BAL, a rendszerető; FENCE, a közlekedés irányítója; ARB, a nagylelkű; FAIL, az örök elégedetlen; SUCCEED, a mindennel megelégedő; végül ABORT, a puccsista. Kicsit szakszerűbben: mindegyik mintaelemként használt változó, s mintaillesztésnél a

— REM felkapja a füzér maradék részét;

— BAL csak a jól zárójelezett kifejezéseket engedi át;

— FENCE szelepként működik: elsőszörre mindent átenged, de a visszalépést már nem engedi meg;

— ARB értéke először az üres füzér, majd hossza fokozatosan nő (vö. a * szerepével a DOS-ban);

— FAIL újabb és újabb alternatíva keresését kényszerítheti ki, mert mindent elvet;

— SUCCEED dinamikus leállást okozhat, mert minden üres füzért elfogad;

— ABORT azonnal berekeszti az illesztési folyamatot, és a hibaágnak adja át a vezérlést.

Az utóbbiak szerepét majd később fogjuk megérteni, egyelőre fogadjuk el, hogy rájuk is szükség van bizonyos szituációkban.

2. Füzérekkel vezérelt függvények

A második csoportba sorolhatjuk azokat a mintagyártó függvényeket, amelyeknek működését argumentumként megadott karakterlánccal vezérelhetjük. Az adott esetben a füzéreken belül az elemek sorrendjének nincs jelentősége, valójában karakterek halmazának megadásáról van szó — füzér formájában. Ezeknek a függvényeknek a legtöbbje már régi jó ismerősünk, nélkülük nem is tudnánk igazán értelmes mintákat készíteni. Párosával járnak, használatuk is gyakran csak ettől eredményes.

SPAN és BREAK kiváló szolgálatokat tettek a sorokból a szavak kiválasztásában: egyikük a szavak elejét tudta gyorsan megkeresni, másikuk a végét. Minden olyan szolgálatra alkalmasak, amikor szét kell trancsírozni a karakterláncokat minőségi szempontok alapján. ANY és NOTANY az előbbiek pontos megfelelői — egyetlen karakterre vonatkoztatva. ANY bármilyen egyetlen karaktert elfogad, amit argumentuma szerint magáénak érezhet, NOTANY listáján viszont éppen azok a „körözött elemek" vannak felsorolva, amelyeket nem szabad elfogadni. Szakszerűbben:

SPAN(str) — felkapja a leghosszabb olyan karakterláncot, amely csak az argumentumfüzérben megadott karaktereket tartalmazza.

BREAK(str) — keresztülengedi a leghosszabb karakterláncot, amely nem tartalmazza az argumentumfüzérben megadott karaktereket, és „lefékez" az első olyan karakterlánc előtt, amely tartalmazza.

ANY(str) — a SPAN egykarakteres változata: elfogad bármilyen „1 hosszúságú karaktersorozatot", de csupán az argumentumfüzér elemei közül.

NOTANY(str) — a BREAK egykarakteres változata. Elfogad bármilyen „1 karakter hosszúságú karaktersorozatot", ha az nem az argumentumfüzér elemei közül való.

3. Egészekkel vezérelt függvények

A harmadik csoportba tartoznak azok a mintagyártó függvények, amelyeknek argumentuma valamilyen egész érték. Ezek használatához van szükségünk a legnagyobb figyelemre, hogy megértsük hasonlóságaikat és különbözőségüket. Az öt függvény közül egy kivételével ezek is párosával tartoznak össze, megérteni ezért tulajdonképpen csak azt a különbséget kell, ami a párokat egymástól megkülönbözteti: az egyik pár szerepköréhez csak az ellenőrzés tartozik, a másikéhoz a begyűjtés. Pontosabban és részletesebben:

LEN(n) — felkap egy n hosszúságú füzért. A hibaágra megy, ha nincs ilyen a „láthatatlan kurzor" pillanatnyi állása mellett.

Ez a függvény igen gyakran jól együttműködik a SPAN-nal és a BREAK-kel, azok ugyanis megállnak bizonyos karakterek előtt — vagy azért, mert azok már nem tartoznak bele a hatáskörükbe, vagy éppen azért, mert beletartoznak. A LEN segítségével ezeket vagy át lehet lépni mint érdekteleneket, vagy éppen hogy fel lehet csípni, és karanténba zárni.

POS(n) — balról ellenőrzi a „láthatatlan kurzor" pillanatnyi helyzetét: sikeres, ha annak pozíciója megegyezik az argumentumban megadott értékkel. POS(0) segítségével biztosítható, hogy csak a füzér elejétől kezdődjék a mintaillesztés.

RPOS(n) — jobbról, a füzér végétől számítva ellenőrzi a „láthatatlan kurzor" pillanatnyi helyzetét. Sikeres, ha a „láthatatlan kurzor" éppen addig jutott el, hogy pozíciója megegyezik a füzér végétől visszafelé számított argumentumértékkel. RPOS(0) segítségével ellenőrizhető, hogy a teljes füzér átjutott-e a mintaillesztés buktatóin.

Figyeljük meg, hogy ennek a két függvénynek csak az igazoltatásra terjed ki a jogosítványa, a begyűjtésre nem! Működésük nem folytatódhat kartanténba zárással, viszont azonnal intézkedhetnek a vezérlés átadásáról, ha valamit nem találnak rendben.

TAB(n) — a „láthatatlan kurzor" pillanatnyi pozíciójától az argumentumban megadott fix pozícióig érő karakterláncot kapja fel. Az üres füzért elfogadja, de a hibaágra kerül, ha a „láthatatlan kurzor" már túlhaladt az argumentumban megadott pozíción, vagy ha az argumentum értéke a mintaillesztés adott pillanatában negatív.

RTAB(n) — a „láthatatlan kurzor" pillanatnyi pozíciójától halad előre a felé a fix pozíció felé, amelyet az argumentumérték a karakterlánc végétől számítva meghatároz. Az üres füzért elfogadja, de a hibaágra kerül, ha a „láthatatlan kurzor" már túlhaladt az argumentum által meghatározott pozíción, vagy ha az argumentum értéke a mintaillesztés adott pillanatában negatív. RTAB(0) alkalmas a hátralévő karakterlánc kiválasztására, RTAB(n) az utolsó n karaktert kapja föl.

Ezeknek a függvényeknek a funkciója viszont elsősorban a begyűjtés. Igazoltatás csak olyankor, ha nyilvánvalóan rendellenes helyzettel találkoznak.

Egy kis ismétlés

A lemezen adok egy kis ízelítőt azokból a mintákból, amelyek eddigi példáinkban előfordultak. Próbáljuk meg kitalálni, melyik mintának mi volt a szerepe! Persze ne feledkezzünk el róla, hogy a fenti, kizárólag mintakészítésre szolgáló függvényeken kívül hellyel-közzel más is szerepel ezekben a kifejezésekben, nevezetesen karanténváltozók és ezek feltöltését biztosító operátorok, egyéb operátorok és függvények, illetőleg a mintával együtt azok a karakterláncok is, amelyekre a mintákat alkalmaztuk.

Floppymellékletünkön egyébként most sem feledkeztem el érdeklődő olvasóink ismereteinek bővítéséről, a felhasznált módszerek részletesebb magyarázatára azonban csak a legközelebbi alkalommal kerítek sort. Kíváncsi vagyok, hogy mi okoz gondot olvasóimnak a program működésének megértésében. Használatához pedig bizonyára jó segítséget nyújt a beleépített rövid összefoglaló.

Vargha Dénes

Clipper 5.01 módosítások

A számítások biztonságáért

A Clipper nyelven írt programoknál egyes hibás matematikai műveleteknél (például LOG(0)-nál vagy 0 hatványozásánál) semmilyen hibajelzést nem kapunk. Ha a hibás értékkel további matematikai műveleteket végzünk, akkor a program az adatállományok lezárása nélkül kilép. Tenni kellett valamit...

A programok futásának adatállomány-lezárás nélküli befejeződését az MSC lebegőpontos számok hibáit figyelő része teszi egy exit utasítással. Mivel makróműveleteknél nem figyelhető a hibát okozó művelet, ezért módosítottam az MSC ezen részét úgy, hogy ne csak a programból való kilépést akadályozza meg, hanem a hibát a Clipper kiírja, és szükség esetén kezelni is tudja.

Mivel a Clipper az MSC matherr függvényét sem használja, ezért a hibák megjelenítésére és kezelésére is írtam egy programot. Az MSC futtató rendszere a matherr függvényt hívja meg a matematikai jellegű hibák felléptekor (például lebegőpontos túl-, ill. alulcsorduláskor stb.). Ha ez a függvény nincs definiálva, akkor az alapértelmezés szerinti hibakezelő lép életbe. A Clipperben ezt egy return(1) utasítással oldották meg, és így az alapértelmezett hibakezelő függvény sem működik.

Ha a CLIPPER.LIB könyvtárállományba akarjuk felvenni az alábbi módosításokat, akkor a

lib clipper -ccalle -DVA -errorsys +math.lib

utasítással tehetjük meg (ha a lib program rendelkezésünkre áll).

Függvények és egyebek

A Clipper matematikai függvényeit a következőkkel bővítettem:

Szögfüggvények:

SIN(x), COS(x), TAN(x), TG(x), COTAN(x), COTG(x), ATAN2(x,y), ASIN(x), ARCSIN(x), ACOS(x), ARCCOS(x), ATAN(x), ARCTG(x), ACOTAN(x), ARCCOTG(x).

A szögfüggvények értékét az MSC szerinti értékhatáron belül számítja és adja vissza. Az értékhatárhibákat a hibakezelő függvények írják ki.

DEG() — A szögfüggvényeket fokban számítja (alapértelmezett), a függvény tetszőleges értéket elfogad, és 90-et ad vissza.

RAD() — A szögfüggvényeket radiánban számítja, a függvény tetszőleges értéket elfogad, és a Pi értékét adja vissza.

ISDEG() — Igaz, ha fokban számolja a szögfüggvényeket.

ISRAD() — Igaz, ha radiánban számolja a szögfüggvényeket.

DEGTORAD(x) — A fokot radiánba számítja át.

RADTODEG(x) — A radiánt fokba számítja át.

PI() — A Pi értékét adja vissza.

Hiperbolikus függvények:

SH(x), SINH(x), CH(x), COSH(x), TH(x), TANH(x), CTH(x), COTANH(x).

Logaritmusfüggvények:

LG(x) LOG10(x) — A tízes alapú logaritmus.

LN(x) LOGE(x) — Az e alapú logaritmus (a Clipper LOG(X) függvényénél jobb hibakezelő tulajdonsággal rendelkezik).

Hatványfüggvények:

POW(x,n) POWER(x,n) — Kiszámítja az x szám n-edik hatványát.

POW2(x) — Négyzetre emel.

POW3(x) — Köbre emel.

POW10(n) — 10-et n-edik hatványára emeli.

NROOT(x,n) — Az x szám n-edik gyökét számítja.

SQR(x) — Négyzetgyököt számol.

CUB(x) — Köbgyököt számol.

Véletlenszám:

RANDOM() — Véletlenszámot ad vissza 0 és 65 535 között lévő értéketben.

SRANDOM(x) — Véletlenszám kezdeti értékét állítja be (0 és 65 535 közti értéket vár).

Átalakító függvények (a lebegőpontos számok tárolását teszik lehetővé az adatállományban úgy, hogy azok a pontosságukból ne veszítsenek):

CTON(n) — A lebegőpontos számot sztringben tárolja, a sztring hossza 8 karakteres mező vagy 8 karakterrel előzőleg feltöltött sztring legyen, például:

```
adat= „12345678"
szam=12.56
adat=ntoc(szam)
? cton(adat)
```

NTOC(s) — A sztringben tárolt lebegőpontos számot visszaalakítja.

A MATHDEMO és SZAMOL programok

A matematikai függvények használatát a MATHDEMO programban példákban adtam meg. A SZAMOL program az adatok bevitelét segítheti elő úgy, hogy különböző számításokat végezhetünk. A programot a következőképpen fordíthatjuk le:

clipper mathdemo

rtlink fi mathdemo,ccalle,dva,matherr,errorsys lib math

A SZAMOL program a matematikai függvények használatát mutatja be közvetlen adatbevitelkor. A programot a következőképpen fordíthatjuk le:

clipper szamol

rtlink fi szamol,ccalle,dva,matherr,errorsys lib math

A mellékelt programok jegyzéke:

CCALLE.OBJ — A C nyelvű kritikus matematikai hibákat kezelő program.

DVA.OBJ — A Clipper 5.01-ben levő matherr program kiiktatását végzi, ennek használatával a matematikai hibák kijelzésre kerülhetnek.

MATHERR.OBJ — A matematikai hibákat C nyelvről a Clipper 5.01 nyelvre átirányítja.

CCALLER.OBJ — A kritikus matematikai hibákat megjelenítő program.

MERROR.OBJ — A matematikai hibákat megjelenítő program.

ERRORSYS.OBJ — A módosított ERRORSYS program, a nullával osztáskor hibát jelez.

MATH.LIB — A matematikai függvényeket és a hozzájuk tartozó hibakódokat tartalmazó könyvtár.

SZAMOL.PRG — Bemutatható program a matematikai függvények használatára közvetlen adatbevitelkor.

MATHDEMO.PRG — Bemutató program a matematikai függvények használatához.

MATH.CH — A MATHDEMO program include állománya.

Forrásprogramok

CCALLE.ASM — A kritikus matematikai hibákat kezeli (a hibakódot a CCALLER-re irányítja).

CFUNC.C — Matematikai függvények (sin, cos stb.).

CUB.C — Előjelhelyes köbgyökvonás.

MATHERR.C — Matematikai hibakezelő (a hibakódot a MERROR-ra irányítja át).

NEWFUNC.C — Lebegőpontos számot átalakítja sztringgé vagy a sztringben tárolt számot visszalakítja (lásd MATHDEMO program).

AFUNC.PRG — Számok kiírását segíti elő.

CCALLER.PRG — A kritikus matematikai hibákat megjeleníti.

ERRORSYS.PRG — A Clipper 5.01 módosított ERRORSYS programja, a nullával osztás kijelzésére.

MERROR.PRG — A matematikai hibákat megjeleníti.

Fürstner János

Életünk bearanyozói a shell scriptek II.

Nyomtatás vezérlése shell scriptekkel

Egy számítógép-hálózatban több központi printer van, amelyekkel ugyan a felhasználók dolgoznak, de működőképességük biztosítása részben a rendszergazda feladata.

A nyomtatással kapcsolatosan az okoz leggyakrabban problémát, hogy a felhasználó kiadja a nyomtatási parancsot, de a nyomtatás vagy el sem indul, vagy a végrehajtás közben „elakad". Az okok általában a következők: nincs online állapotban a printer, nyomtatás közben kifogy a papír, a nyomtató begyűri a papírt, hálózati hiba jelentkezik stb. Ha a rendszergazda nem tartja kézben a nyomtatási folyamatot, akkor egy idő után a nyomtatók spool területein a nyomtatók várakozó soraiban (a printer queue-kban) felgyűlhetnek — az esetleg már rég aktualitásukat vesztett — jobok. Ennek a következményei közt meg kell említeni a háttértár indokolatlan foglalását, valamint esetleg azt, hogy amikor a nyomtatás leállását kiváltó ok megszűnik, a nyomtató olyan nyomtatási parancsokat is végre fog hajtani, amelyekre talán már nincs is szükség.

A Unixot a nyomtatási sor vezérlésével kapcsolatban jól ellátták lehetőségekkel, többek közt rendelkezik az lpq (line printer queue) és az lpc (line printer control) parancsokkal is. Azonban — bár a feladat velük is megoldható — egyikükkel sem lehet egy, a nyomtatási sorokra vonatkozó célvizsgálatot kulturált módon, jól áttekinthető kimeneteket kapva végrehajtani. Magyarul ez a következőt jelenti: ha meg akarjuk állapítani, hogy a hálózatunkban mely printerek azok, amelyekkel esetleg éppen „gond van", akkor célszerű erre a célra egy parancsfájlt írni. Most pedig — az e cikk előző részében már megismert tematikával — nézzük meg a feladat egy lehetséges megoldását.

A Unix a nyomtatókkal kapcsolatos technikai információkat egy állományban, a /etc/printcap-ben tárolja. Az állomány számunkra lényeges részei az alábbiak.

```
PARANCS:
cat /etc/printcap
OUTPUT:
# @(#)printcap 4.1 (ULTRIX) 7/2/90
lp|lp0|0|matrix:\
.
.
        :sd=/var/spool/lpd:\
.
.
lp1|1|laser1:\
.
.
        :sd=/var/spool/lpd1:\
.
.
lpn.........
```

A verziószám stb. után láthatjuk, hogy az állomány a printerek szerint tagolódik. Minden szövegrész egy fejléccel kezdődik, ahol „|" pipe-karakterek választják el a mezőket. Az első mező a printer neve. Az alatta levő sorokban valahol megtaláljuk az sd (spool directory) változót is. Az operációs rendszer ennek a változónak az értékéből „tudja", hogy egy nyomtatási parancs kiadása után hová kell küldenie a kinyomtatandó állomány másolatát. A másolatot innen az ún. lpd démon próbálja továbbítani a nyomtatóhoz. Ha ez nem sikerül az első próbálkozásra, akkor időnként újabb kísérletet tesz. Az állomány másolata mindaddig ott marad, amíg az lpd démon sikerrel nem jár, vagy pedig nem töröljük.

Fentiek ismeretében most már mi is tudjuk, hogy a rövidesen elkészülő parancsfájllal hol vizsgáltassuk a nyomtatásra várakozó állományok jelenlétét. Nézzük meg, mit találunk egy ilyen könyvtárban, például az lp1 nyomtatóéban.

```
PARANCS:

ls /var/spool/lpd1

OUTPUT:

/var/spool/lpd1/cfA460abcdefg
/var/spool/lpd1/dfA460abcdefg
/var/spool/lpd1/init
/var/spool/lpd1/lock
/var/spool/lpd1/status
```

Készülő programunknak az első két sorban szereplő állományok valamelyikére van szüksége. (A nyomtatandó állomány másolata a dfA*, a cfA* pedig az lpd démon egy munkafájlja. A Unix egyszerre kreálja, majd pedig szünteti meg őket, így megoldandó feladatunk szempontjából egyenrangúak.) Az állományok neveiben a cfA, illetve a dfA részek állandóak, az utánuk következő számok sorszámok, amelyek minden nyomtatásnál az ugyanitt található lock fájlból aktualizálódnak, míg az utolsó részben abcdefg-vel jelöltük a gazdagép nevét.

Mivel a nyomtatók alkönyvtárai szerencsére mind a spoolból ágaznak le, és mindegyik neve lpd-vel kezdődik, kiadhatjuk az alábbi parancsot.

```
PARANCS:

ls /var/spool/lpd*/cfA???abcdefg

OUTPUT:

/var/spool/lpd1/cfA460abcdefg
/var/spool/lpd5/cfA489abcdefg
/var/spool/lpd5/cfA490abcdefg
```

Láthatjuk, hogy az egyes printerek várakozó sorában van-e nyomtatásra váró állomány. Az lp1 printerre egy, míg az lp5-re két állomány várakozik.

A kitűzött célt már majdnem elértük, azonban az igazán jó az lenne — mivel bennünket valójában nem érdekel sem a várakozó állományok neve, sem pedig mennyisége, csak az, hogy vannak-e vagy nincsenek —, ha a program csak a kívánt információkat írná ki.

```
PARANCS FILE:

#
echo " "
echo "Az alábbi printerek várakozó sora
nem üres :"
echo " "
( ls /var/spool/lpd*/cfA???abcdefg >
~/queue.lst ) > &~/quv_error.txt
awk ' BEGIN           { FS="/" }\
                      { lp_max=20 }\
   $4 == "lpd"        { lp[0]="lp0" }\
   $4 == "lpd1"       { lp[1]="lp1" }\
   $4 == "lpd2"       { lp[2]="lp2" }\
   $4 == "lpdutolso"  { lp[lp_max]="lp-utolso"
}\
   END  { for ( i=0;i < lp_max+1;i++ )\
        printf "%s ",lp[i] } ' ~/queue.lst
echo " "
echo " "
rm ~/queue.lst
```

Az első sorban levő „#" jel utasítja a Unixot, hogy a parancsfájl értelmezéséhez a rendelkezésre álló shellek közül a cshellt válassza. Ez egyben azt jelenti, hogy a „#" jellel kezdődő shell scripteknek a cshell szintaxisának kell megfelelniük.

Az ls utáni rész egyrészt az ls kimenetét, másrészt a hibaüzeneteket irányítja át egy-egy állományba. A „~" tilde karakter az ún. home directory jele, azaz a kimeneti és a hibaállomány is minden felhasználó saját könyvtárában fog keletkezni. A „&" jel a diagnosztikai kimenetet jelzi. Mint láthatjuk, az utasítások zárójelezhetők.

Az awk nyelvet a cikk előző részében már elkezdtük kóstolgatni, most folytathatjuk. Maga az awk program az „awk" szó utáni „ ' „ aposztróf jellel kezdődik, és a (záró) aposztrófjelig tart. A BEGIN és az FS már ismerősök, nézzük a többit.

Az egyes sorok végén található „\" jel az ún. folytatósor jele. Olyan helyeken kell alkalmazni, ahol egy utasítást több sorra szabdalunk fel. Egy awk program a bemenetéül szolgáló állományt soronként elemzi, a bemeneti állomány minden sorára végrehajtódik — a BEGIN és az END kivételével — az egész program.

A ' $4 == „lpd" ' és hasonlóak, feltételes utasítások feltételrészei, amelyek teljesülése esetén végrehajtódnak a kapcsos zárójelek közötti utasítások. Egy ilyen sor tehát például azt jelenti, hogy ha a bemeneti állomány aktuális sorának negyedik mezője (ebben a speciális esetben, mivel a sorok a szokásostól eltérően „/" jellel, azaz mezőszeparátorral (FS=„/") kezdődnek, nem a negyedik mezőről, hanem a negyedik mezőszeparátor előtti mezőről beszélhetünk) az „lpd" szöveggel egyenlő, akkor a program végrehajtja az utána következő értékadó utasítást. Ha nem volt egyenlő, akkor a program — a bemeneti állománynak ugyanarra a sorára vonatkozóan — kiértékeli a következő feltételt stb. (Működése hasonló a C nyelv switch utasításához.)

Az END a BEGIN ellentéte, az utána következő utasításokat csak egyszer hajtja végre a program, mégpedig a teljes bemeneti állomány elemzése után. A hozzá tartozó ciklus nem szorul magyarázatra.

A záró aposztróf jel utáni állománynév adja meg az awk programnak a bemeneti állományt. Az utolsó sorban a rm utasítással töröljük a (bemeneti) munkaállományt.

```
OUTPUT:

Az alábbi printerek várakozó sora nem
üres :

lp1    lp5
```

A program tehát eleget tesz annak a kívánságunknak, hogy kényelmesen kaphassunk áttekinthető információt a printerek várakozó sorairól. Azt, hogy a már „felderített" várakozó sorokkal hogyan lehet shell scriptek segítségével bánni, a cikk következő részéből megtudhatjuk.

Losonczi János

Gyorsítsa hálózatát tízszeresére.
Mi a hét éves hálózatépítési tapasztalatunkat adjuk hozzá.

AMI ENTERPRISE III VESA
Local bus alaplap
(486/66 MHz/256 K)
+ AMI FAST DISK EISA SCSI-II
Caching adapter
+ TCNS EISA koax hálózati csatolókártya

TCNS

VAX teljesítmény PC áron NOVELL hálózaton!

Nagy teljesítményű PC alaplapok és I/O perifériák (American Megatrends Inc.)

AMI

THOMAS•CONRAD®
CORPORATION

A VILÁG LEGGYORSABB HÁLÓZATA!

TCNS *100 Mbit/sec* sebességű hálózati rendszerek!

A mi célunk, hogy Ön célba érjen!

MEGATREND

6000. Kecskemét, Szarvas u. 24.
Tel./Fax: (76) 326-290
1201. Bp, Külső Török Flóris u. 24.
Tel./Fax: 147-9094

1173. Budapest, Pesti út 8-12.
Tel./Fax: 158-7544, 158-7727, 158-7500, 158-7100, 158-7599
(14-es, 41-es mellék)

INFORMÁCIÓKÉRÉS: A0470 ▶

BUYER'S GUIDE

KATALÓGUS '93

A számítástechnikai piac iránytűje

Már csak a kiadónál rendelheti meg a **KATALÓGUS '93**-at, a számítástechnikai piac iránytűjét.

A kiadvány

* táblázatokba foglalja a hardverek, szoftverek és szolgáltatások jellemzőit és árait;
* bemutatja a hazai fejlesztésű szoftverek legteljesebb választékát;
* segítségével az azonos termékek többféle szempontból összehasonlíthatók.

A tavaszi KATALÓGUS '93
megrendelhető közvetlenül a kiadótól,
ára 313 forint + postaköltség.

IDG Magyarországi Lapkiadó Kft.
Terjesztési Osztály
1012 Budapest, Márvány u. 17. VIII. emelet
Postacím: 1536 Budapest, Pf. 386
Telefon: 156-3211/235 vagy 156-0337/12

INFORMÁCIÓKÉRÉS: A0558 ▶

ÚJ ALAPLAP KÖNYVEK!

Jodál Endre

Informatikai alapszókincs

Angol–magyar szótár

15 000 címszó, ára: 356,– Ft

Csórián Sándor

Számítógépes kommunikáció

Hálózatokról felhasználóknak és döntéshozóknak

Az SZKI támogatásával, ára: 356,– Ft

CÉDRUS Kiadó Kft
1441 Budapest VIII., Kőbányai út 21.
Tel./Fax: 269-9128

INFORMÁCIÓKÉRÉS: A0482 ▶

KC

Konkoly Computer Szaküzlet

Bp. V. Nádor u 19. Tel.: 131-9166
18 óra után üzenetrögzítő.

HEWLETT PACKARD Számítógépek és tartozékaik

Lézer és tintasugaras nyomtatók
Színes és monó lapszkennerek

ALR — Advanced Logic Research, Inc.

The Macro

9 és 24 tűs nyomtatók
mono ill. színes változatban.
Lézernyomtatók
PostScript interface,
memóriabővítések, tonerek,
lapadagolók

Alaplapok (386-486)
Memóriák
Video adapterek
Egerek, Handy szkennerek
Merevlemezek (40MB-1GB)

star

MPC
Multimedia PC

Multimédia rendszerek
CD ROM meghajtó

Sound Blaster Hangkártyák
MIDI interface

SB SOUND AUDIO BLASTER

INFORMÁCIÓKÉRÉS: A0354 ▶

A Mikrobazár rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hirdetéseket közlünk.

A kereskedelmi tevékenységet szolgáló apróhirdetések tarifája gépelt soronként (60 karakterenként) 300 Ft. Kérjük, hogy a hirdetés díját az IDG Lapkiadó Kft.-nek a Magyar Külkereskedelmi Banknál vezetett, 203-28016 számú számlájára utalják át, vagy postautalványon fizessék be az IDG címére (1536 Budapest, Pf. 386), a hátoldalon feltüntetve, hogy „Alaplap apróhirdetés". A befizetést igazoló szelvényt a közlendő hirdetési szöveggel együtt az Alaplap szerkesztőségéhez (a kiadóéval azonos címre) küldjék el.

A nem kereskedelmi célú egyéni hirdetések közlése INGYENES!

Amiga programok nagy választékban eladók. A programokról lemezen listát küldök. Ugyanitt DS/DD lemezek eladók. Cím: Dikó István, 1053 Budapest V., Veres Pálné u. 9. Tel.: 137-3193.

Eladó **Amiga 500-hoz** Kickstart 2.0 bővítő, dobozolt 86 MB (AT-BUS) HD, illesztőpanel. Ára: 48 000 Ft. Ugyanitt Amiga 500 alapgép (1 MB RAM, Bootselector, HI PWR tápegység) 38 000 forintért eladó. Egy DigiView kompatibilis Video-digitiser 9000 forintért szintén vevőre vár. Cím: Kovács László, 2403 Dunaújváros, Pf. 15.

Eladó **Atari 800 XL** alapgép. Az árban megegyezünk. Cím: Károly Márton, 9700 Szombathely, Bertalanffy u. 69. Tel.: (94) 26-300.

Enterprise programok eladók. Válaszboríték ellenében listát küldök. 2500 program, sok kedvezmény, ajándék. Cím: Zemen László, 1164 Budapest XVI., Olló u. 16.

Enterprise programok eladók. Válaszboríték ellenében listát küldök. 2000 program, sok kedvezmény, ajándék. Cím: Zemen László, 1164 Budapest XVI., Olló u. 16.

Eladó **Enterprise 128**: 360 kB floppy, illesztő, monitor- és nyomtatókábel, valamint sok más tartozék. Cím: Gémes György, 3908 Rátka, Iskola tér 9. Tel.: Rátka 49.

C-64-es programok kazettán eladók. Válaszboríték ellenében a 6700 programról listát küldök. Cím: Bohács Tibor, 4320 Nagykálló, Petőfi u. 8. Tel.: (42) 63-389.

C-64-es programok kazettán eladók. Válaszboríték ellenében listát küldünk. Cím: Sándor & Szőke, 3073 Tar, Május 1. út 8.

Eladó **C-64-re** a legújabb, a legtöbb kényelmi funkcióval ellátott katalóguskészítő program. Cím: Borbély Balázs, 1116 Budapest XI., Sáfrány u. 44. Tel.: 162-7724.

Eladók felhasználói, játék- és demoprogramokkal teli **5,25"-os HD lemezek** (100 Ft/db). Cím: A-BOX, 1399 Budapest, Pf. 701/783.

Eladók **IBM játék- és felhasználói programok**. Áruk 40 Ft lemezenként, HD-s lemezek esetében 100 Ft. Cím: A-BOX, 1399 Budapest, Pf. 701/783.

Eladók **IBM AT/XT programok** nagy választékban. Több ezer program közül választhat! Ugyanitt DS/HD lemezek eladók. Cím: Dikó István, 1053 Budapest V., Veres Pálné u. 9. Tel.: 137-3193.

Eladók új **játék-, demo- és felhasználói programok**. Válaszboríték ellenében listát küldünk. Cím: Amiga Friend Two Crew, 1399 Budapest, Pf. 701/55.

Olcsó eladók IBM PC-re **programok**. Cím: Bagi László, 5130 Jászapáti, Berdó u. 9.

Írjál vagy telefonálj, ha Téged is érdekelnek az új vírusmentes **játékok** IBM PC-re. Kérésedre listát küldök. Cím: Botyánszki Péter, 5600 Békéscsaba, Derkovits sor 3. Tel.: (66) 327-040.

Kétkulcsos ÁFA program eladó. Irányár: 5000 Ft utánvét. Cím: Svantek Andrea, 2600 Vác, Radnóti M. u. 29.

Megrendelhető a **Clipper 5.01** objektumkészítő új verziója. Külön kérésre tájékoztatót is küldök. Cím: Szűcs János, 4400 Nyíregyháza, Vasvári Pál u. 37. IV.13. Tel.: (42) 13-568 vagy (42) 12-222 / 1382-es mellék.

PC szoftver akció! Eladó Turbo Pascal 7.0 — 5 000 Ft, felirattervező program — 1000 Ft, angol szótár — 500 Ft, ChiWriter — 2000 Ft. Cím: Lovas Zoltán, 4400 Nyíregyháza, Toldi út 68. VII./57. Tel.: (42) 14-253 (délután).

Eladó **20 MB-os Seagate** winchester (ST-225), ára: 9000 Ft. Cím: Vargha Dénes, 1061 Bp. VI., Andrássy út 32. Tel.: 131-4082.

Eladó Scan típusú CGA monitor. Cím: Jakubovics Demeter, 1131 Budapest XIII., Keszkenő u. 5.

Eladó IBM PC-hez **Covox hangmodul** 750 forintért, **epromégető** 7500 forintért. Ugyanitt **MCS48 és MCS51 mikrogép-fejlesztő rendszerek** 10 000 forintért eladók. Válaszboríték ellenében részletes ismertetőt küldök. Cím: Mikroklub — Torkos Csaba, 8100 Várpalota, Pf. 65. Tel.: (80) 71-439.

Számítástechnikai oktatás IBM PC gépen bármilyen témában! Beszerzési tanácsadást és programkészítést is vállalok! Cím: Fridl György. Tel.: 162-2070 (csütörtökön 16-18 óra között).

Vennék jó állapotban lévő AT 286-os számítógépet max. 30 000 forintért: 40 MB HDD, 1,2 MB FDD, 101 gombos billentyűzet, Hercules monokróm monitor, 1 soros, 1 párhuzamos port. Cím: Pásztor Zsolt, 5350 Tiszafüred, Esze T. u. 4./a.

Keresek IBM PC-re animációt támogató felhasználói szoftvereket és játékprogramok írásához kifejlesztett programozási nyelveket. Cserébe több száz programról listát küldök. Cím: Miszkulyi Sándor, 2085 Pilisvörösvár, Külső Bécsi út 263.

E számunk hirdetői

Bajnok kerestetik

A Magyarországon is népszerű DataFlex alkalmazásfejlesztő rendszer nemcsak DOS, Windows és Novell Netware alatt üzemel, hanem működik az Intel 386, 486, a Motorola 88000 és az IBM RISC 6000 processzoros Unix operációs rendszerek alatt is. A ma kapható legolcsóbb 32 bites operációs rendszerhez is — az OS/2 2.0-ás verziójához — kifejlesztették a DataFlexet. A szoftver 3.03-as verziója egy valódi objektumorientált negyedik generációs programnyelvet és egy relációs adatbáziskezelőt tartalmaz. A DataFlex objektumokkal fejlesztett programok egységes kezelési felületet nyújtanak, és megfelelnek az OS/2 alapjául szolgáló IBM SAA/CUA szabványának. A 32 bites OS/2 alatt futó szoftverrel egy asztali számítógépen percenként akár 60 000 rekordművelet is elvégezhető.

Az objektumorientált és 4 GL programozás népszerűsítése érdekében programozóbajnokságot hirdet a DataFlex, a NJSZT DataFlex Klub és a disztribútori feladatokat magára vállaló Pentacomp Kft. A június 18-19-én megrendezendő erőpróbán azt a legjobb magyar DataFlexp-rogramozó csapatot szeretnék megtalálni, amely az Európa-bajnokságon képviselhetné Magyarországot. A 14 órán keresztül zajló versenyen a 3 fős csapatoknak a verseny kezdetekor átadott esettanulmányt kell kidolgozniuk saját hardver- és szoftvereszközeikkel.

Teljesen felfrissülve

Közel egy éve egyedi Cocom-engedély nélkül szerezhetők be Magyarországon a RISC/6000-es gépek. Az elsősorban számításigényes feladatokra és grafikai alkalmazásokra használt gépcsalád 9 új taggal gyarapodott.

Az X-terminál alternatívájának tekinthető legkisebb modellnél (M20) a processzort integrálták a monitorba. A grafikus munkaállomások piacán ez a modell rendelkezik a legkedvezőbb ár/teljesítmény mutatóval, szerverkategóriában pedig az 570-es modell nyújt különösen kedvező ár/teljesítmény viszonyt. Eddig konfigurációtól függően lehetett ugyanaz a gép munkaállomás vagy szerver. Most 3 modellt (355, 365, 375) speciálisan munkaállomásnak konfigurálták, amelyeket elsősorban mérnököknek és kutatóknak ajánlanak.

A bejelentés kapcsán a már meglévő modellek ára 12-25%-kal csökkent. Az újonnan bemutatott gépeknél megnőtt a felhasználók szabadságfoka: ki-ki maga döntheti el, hogy erős grafikával és kevésbé erős CPU-val (vagy fordítva) akarja-e az immár nagyobb tudású és kedvezőbb árfekvésű gépét konfigurálni.

Rendszerben gondolkodva

Ha IntRam, akkor hardver — ez az asszociációja a hazai számítástechnikusok legtöbbjének. Ezen a sematikus képen igyekezett változtatni az IntRam azon a Béke Szállóban megrendezett kiállításon, amelyre partnereit (Aerus, Eurogold, Keszo) is meghívta. A szoftveres cégekkel való együttműködés eredményeként valamennyi számítógépet vírusvédő rendszerrel látják el (Sysdoki). Az Eurogold által fejlesztett moduláris felépítésű integrált ügyviteli rendszert (Érték) az IntRam elsősorban vegyesvállalatoknak ajánlja. A vezetői információs részben a felhasználó saját szempontjai szerint tölti ki a keretrendszert. A C-ben és Clipperben írt szoftverhez „még számítógépre sincs szükség" (!), mert a programhoz ajándékba kapnak egy 286-os Olivetti számítógépet. A kiállításon az IntRam — mint Olivetti-dealer — az olcsó otthoni számítógépektől a csúcsminőségű professzionális típusokig az Olivetti számítógépek és irodatechnika széles választékát mutatta be. A számológépek, menedzserkalkulátorok és írógépek mellett a pénztárgépek széles skáláját is felvonultatta — ez utóbbihoz már fejlesztik a szoftvert is. Egy eddig tisztán hardveresnek tekinthető cég filozófiája alakul át úgy, hogy nemcsak „vasban", hanem rendszerekben (is) gondolkodnak...

Több benne a szoftver

Az eddig elsősorban hardverforgalmazóként számontartott Albacomp a hazai CAD/CAE/CAM-világban is meg akar erősödni. Szerződést kötött az igényes grafikai megoldásairól ismert Intergraph céggel, hogy unixos és PC-s komplett alkalmazásait forgalmazza. Unixos demórendszert nemcsak Székesfehérvárott láthattunk, hanem az Ifabón is bemutatták térinformatikai (városirányítás, közművek, térképészet, képfeldolgozás), mechanikai rendszerekhez kapcsolódó és kultúrmérnöki alkalmazásait.

A térinformatika (GIS) csúcstermékének számító MGE rendszer talán a legismertebb Intergraph szoftver, amely a grafikus és alfanumerikus adatok előállítására és menedzselésére szolgál. A nagy sikerű Dynamo automatizálja a térinformatikát: kiszámítja és ábrázolja, hogy egy új út tervezésekor hány házat kell lebontani, honnan kell embereket átköltöztetni. A manapság rendkívül aktuális városirányítási rendszerek és közműnyilvántartások mellett a kultúrmérnöki alkalmazások (út-, vasút- és hídtervezésre alkalmas programcsomagok) iránt is nőtt a kereslet: speciális vízügyi tervező rendszereket kombinálnak településfejlesztésre alkalmas szoftverekkel. A tervezéstől a késztermék előállításáig olyan komplett mechanikai programcsomagok gondoskodnak, amelyek egységes keretrendszerbe foglalják a CAD/CAM alkalmazásokat.

Printer, plotter — nyomott áron

A Hewlett-Packard perifériák eddig is jó minőségükről és nagy megbízhatóságukról voltak nevezetesek. Printerek és plotterek piacán a HP további babérokra vágyik: egész sor új perifériát jelentett be — kedvező áron.

A tintasugaras technológiával működő nyomtatók közül két új típus jelent meg a hazai piacon: a DeskJet 510 és a DeskJet 1200C. A 300 dpi felbontású 510-es modell feltehetőleg a DeskJet 500-at váltja ki, 40 000 forintos dealeri áron. A lézerprinterek közül a LaserJet 4L és a LaserJet 4Si számít újdonságnak. A 4L 300 dpi felbontású nyomtatóval — amelynek semmi köze sincs a LaserJet 4, illetve 4M printerekhez — a HP valósággal hadat üzen a mátrixnyomtatóknak. Igaz, hogy nem produkál egyszerre 3-4 oldalas másolatot, így számlakészítésre nem alkalmas, azonban 90 000 forint körüli árával jelentős piaci tényezővé válhat.

A HP plotterkínálata két új típussal gyarapodott: DesignJet 650 C és a DraftPro Plus rajzgépekkel. A DesignJet 600 utódjának tekinthető 650-es színes, tintasugaras plotterrel 300 dpi-vel, 256 színnel rajzolhatunk. A 4 MB memóriájú rajzgép 750 000 forintos dealeri áron kapható, míg a 8 tollas, A0-A4 méretig rajzoló DraftPro Plus-ért 484 000 forintot kell fizetni a viszonteladóknál.

Sziebig Andrea

Nemvagyunkhibátlanok

Vállaljuk, „kertelés nélkül”

Létraversenyünk új feladatai a múlt hónappal véget értek. A megoldások ismertetésével és kiegészítő megjegyzésekkel azonban a rovat tovább folytatódik. Mint ígértük, júliusban sor kerül a verseny végső eredményeinek közzétételére. Hogy a jutalmak mik lesznek, az egyelőre még maradjon titok. De már nem sokáig...

Mentegetődzéssel kell kezdenem. Forrásomat a szilveszteri rumliban nem ellenőriztem kellő alapossággal, így néhány pontatlanság benne maradt a feladatban. Olvasóink becsületére legyen mondva, hogy a lényeges jelenségeket ők ennek ellenére kihámozták a feladatból.

1. A szövegben van egy elírás: „manda” áll mande helyett, pedig a „mande” — „gaode” szembeállításnak megvan a maga szerepe a megoldásban.

2. A „man” — „wan” szembeállítás félreértés folytán került a szövegbe a sokkal természetesebb „kuai” — „man” helyett. Ugyanis éppen a „man” jelenti a „lassú”-t, ennek ellentéte pedig a köznapi nyelvben „kuai”. (A félreértés oka: létezik egy „mang” — „wan” szembeállítás is, „sietősen” — „késlekedve” értelemben, de az sokkal kevésbé illik ide.) Ezúton is elnézést kérek e „félrevezetésért” minden jelenlegi és jövőbeli sinológustól.

A mélyvízben

Szabó Judit azzal kezdi megfejtését, hogy különböző szempontok szerint csoportosítja az odaillő mondatokat, és különböző megfigyeléseket tesz. Először a tőmondatokat veszi szemügyre:

1.1 Ta chi fan: Ő eszik.

1.2 Ta kan shu: Ő olvas.

1.3 Ta xie zi: Ő ír.

Már itt meglepődik, hogy két-két magyar szónak három-három szó felel meg a kínaiban. A meglepetés akkor kezd átmenni megértésbe, amikor rátalál a „fan” és a „zi” szavakra más típusú mondatokban. Nosza, végezzük el a csoportosítást ezekre a birtokviszonyos mondatokra is!

2.1 Zhe shi tade fan, tade zi he tade bao : Ez az ő rizse, betűjele, újságja. A „tade” háromszoros előfordulása árulkodó: ez lehet a birtokos névmás.

2.2 Chi gaode nüren shi tade muqin: Az az asszony, aki a buktát eszi, az anyja.

2.3 Xie ci xie de mande (! jav. V.D.) ren shi tade fuqin: Az az ember, aki lassan (!) írja a szót, az apja.

Ezekből nagy valószínűséggel kihámozható a mondatok második fele: „van övé anya” és „van övé apa”. De feltűnik az is, hogy a „tade” szó az egyszerű mondatokban szereplő „ta” szó származéka lehet.

A következő csoportosításban a határozószós mondatok kerülnek sorra, részben szembeállítva egyszerű mondatbeli megfelelőikkel, majd az igék jelentése szerinti csoportosítás. Ezekután Szabó Judit az alábbi három szerkezettípus alapján összegzi megfigyeléseit:

„1. eset: Ha a mondatban nem szerepel a tárgy, vagyis hogy mit eszik, olvas vagy ír az illető, akkor az »eszik« ige »chi fan«, az »olvas« »kan shu«, az »ír« pedig »xie zi« lesz. Biztosan azért, mert a kínaiak általában rizst esznek, könyvet olvasnak, és betűt írnak. Ha éppen ezekre az általános tárgyakra gondolunk, akkor is ezt a formát kell használni.

2. eset: Ha a mondatban van tárgy, és az eltér az előzőekben emlegetett általános tárgytól, akkor ezt az általános tárgyat helyettesíteni kell az új tárggyal (pl. buktát eszik, szót ír).

3. eset: Ha a mondatban van határozószó, a képzés szabálya a következő: IGE + TÁRGY + IGE, vagyis a fenti módon meg kell kettőzni az igét.”

Itt most félbeszakítjuk olvasóink megfejtéseinek ismertetését, mégpedig nagyon egyszerű okból. Nem akarjuk hátrányba hozni azokat az olvasóinkat sem, akik esetleg a fenti elírás miatt nem küldték el megfejtésüket. Az ő kedvükért erre a feladatra még június 1-jéig elfogadunk megfejtéseket, csak kérjük, hogy vegyék figyelembe a fenti javításokat. Az eddig beküldött megfejtések értékelésénél természetesen nem a megfejtő hibájának tekintjük, ha problémái támadtak a „mande” szó hibásan írt alakjával, sőt! Javítást csak akkor küldjenek be, ha lényegesen módosult az álláspontjuk a fentiek következtében.

Tartozom azonban egy további kiegészítéssel is, ami nem érdektelen a megfejtés szempontjából. Arról van ugyanis szó, hogy

akínaiakmindentegybeírnak

A kínai szövegekben nincs szóköz, sőt sajnálatos módon a mai napig sincs egyértelmű megállapodás arról, hogy latin betűs átírásban miként tagolják a szöveget. A szavakra tagolás szempontjából az egyik kényes kérdés, hogy mit csináljanak azokkal a szócskákkal, amelyek régebben önálló lexikai jelentéssel bírtak, és most kezdenek átmenni tisztán grammatikai szerepű segédszavakba. (Az egyiknek az eredeti jelentése például valami olyasmi volt, hogy „mégpedig”.)

Feladatunkban két ilyen szócska is akad, bár kiejtésben és latin betűs átírásban ezek nem különböznek egymástól. Funkciójuk azonban nem teljesen ugyanaz, sőt hieroglifájuk sem. Még azt is elárulom, hogy bizonyos átírási szokások szerint az egyiket az előtte lévő szócsoport végére szokták ragasztani.

Akárhogy igyekeztem, nem sikerült hieroglifájuk elemeiből sem rájönni, hogy milyen gondolattársítás húzódhat meg eredeti jelentésük mögött. Az egyik szócska hieroglifája ugyanis könnyen dekódolható: azt jelenti, hogy „fehér kanál”. Csak azt nem lehet tudni, hogy miért. A másikban pedig a „lépés”-nek, a „nap”-nak és a „gyerek”-nek az elemeit vélem felfedezni. (Persze mit lehet tudni egy olyan nyelvben, ahol a „kertelés nélkül” helyett azt mondják: „kinyit kapu, meglát hegy”?

A határidő módosítása miatt a februári megfejtések értékelését egyelőre felfüggesztjük, hogy majd a végleges számíthassuk hozzá mindenkinél. Jövő számunkban a márciusi eredményeket ettől függetlenül közzétesszük. Addig is várjuk az újabb megfejtéseket.

Vargha Dénes

A tesztek tesztelése

Memorizálás és installálás

Érdeklődve olvastam a PC-World februári számában — a Tippek és tanácsok rovatban — a „64 kilobájt nyereség" című cikket. Ezzel kapcsolatban szeretném megosztani tapasztalataimat az Alaplap olvasóival, majd pedig az installálási procedúrát egyszerűsítő ötleteimet adom közre, hátha másoknak is hasznosak lehetnek.

A PC-World valamely korábbi számában összehasonlító tesztet olvashattunk számos memóriamenedzser-programról, köztük a QEMM386-osról is. Az említett cikkből is kitűnt, hogy a QEMM386 6.01-es verziója minden paraméterében és tudásában a legjobb. Azon túl, hogy teljes mértékben kompatibilis a VCPI és a DPMI szabványokkal, takarékosabban bánik a memóriával és jobban használhatóvá is teszi azt, mint a HIMEM.SYS és EMM386.EXE páros.

A HIMEM.SYS és az EMM386 ráadásul nem is teljesen kompatibilis a DPMI szabvánnyal (ami ugyebár a Microsoft háziszabványa). Minderre akkor kellett rádöbbennem, amikor a Windows 3.1 alatt futtatva a Borland Pascal 7.0-ás csomag új debuggerét, az teljesen kiakasztotta a HIMEM.SYS-t. Ennek csak annyira futotta, hogy kiírta egy sorba: egy illegális DPMI-hívás zavarta össze a lelkivilágát, majd — mint aki jól végezte dolgát —, visszaadta a DOS-nak a vezérlést!

Mindezt elkövette a Windows 3.1-hez és az MS-DOS 5.0-hoz adott HIMEM.SYS is. A fenti hibajelenség viszont soha nem jelentkezett, ha a QEMM386-ost használtam mint memóriamenedzsert.

Szintén gondot okozott a februári cikkben említett „NOEMS" opció is. Számos szoftver, amelyeket nem a Windows alá írtak és használják a VCPI ajánlásokat követő PHAR LAP extendert, nem hajlandóak futni, ha a fenti opció be van állítva (például az AutoCAD).

Mit tegyünk hát, ha a fenti szoftvereket használni szeretnénk, de szükségünk van 92 kbájtnál nagyobb felső memóriaterületre is?

Két lehetőség

1. Készítünk annyi és olyan CONFIG.SYS állományt, hogy a különböző szoftverekhez valamelyik mindig illeszkedjen, és gondoskodunk arról, hogy mindig az legyen aktív, amelyiket éppen használni kell.

2. Beszerzünk egy memóriamenedzser-programot (például a QEMM386-ost), és csak ezt írjuk be a CONFIG.SYS állományba, megfelelően paraméterezve.

Az első lehetőség a hibák egy részét megoldja ugyan, de rendkívül fárasztó és nehezen követhető, hogy éppen melyik beállítás aktív. (És elég bosszantó, ha éppen nem az, amelyiket mi szeretnénk.) Tehát csak a második nyújt jó megoldást, bár itt is igaz: ne válasszuk a „NOEMS" opciót, ha VCPI-s szoftvert használunk!

A korábban említett cikkben szó van arról, hogy a QEMM-en kívül számos egyéb program is lehetőséget ad a memória jobb kihasználására (például a 386MAX is). Jómagam mégis a QEMM mellett maradtam máig is, az alábbiak miatt.

A hűség előnyei

Először a QEMM386 5.11-es verziójával találkoztam, majd szemügyre vettem a többieket — például a 6.01-es verziót —, de akárhánnyal próbálkoztam, mindig visszatértem az 5.11-eshez. Összefutottam a 386MAX-szal is. Szimpatikus volt rendkívül fejlett telepítő és optimalizáló programja miatt, mégis a többiek sorsára jutott. Rövid ismerkedés és tesztelés után kitöröltem a CONFIG.SYS-ből, de meghagytam a cache-programját (erről lásd később).

Miért kanyarodtam vissza mindig az 5.11-eshez? Három fő okból:

1. Az új memóriamenedzser túl nagy helyet igényel az extended memóriából. A QEMM386 6.01 és a 386MAX 6.0 is 256 kbájtot foglalt le a saját maga számára.

2. Nem minden program volt hajlandó elfogadni a jelenlétét vagy az általa definiált memóriatípust.

3. Körülményes volt a felső memória elérése.

Az első ok az én esetemben azért kapott nagyobb hangsúlyt, mert az a gép, amellyel dolgozom, csak 2 Mbájt RAM-ot tartalmaz, és ha abból vesz el 256 kbájtot a memóriamenedzser, akkor a Windows 3.1 csak real módban hajlandó elindulni. A QEMM386 5.11 pedig beéri annyival, mint a HIMEM.SYS, azaz 64 kilobájttal, mégpedig úgy, hogy ő is befér a felső memóriába a DOS mellé.

A második ok: minden program gond nélkül fut vele, mert attól függően, hogy az adott program milyen memóriát igényel (EMS-t vagy XMS-t), a QEMM mindig azt nyújtja, amire szükség van.

A harmadik ok: az optimalizáló program már az installálás során gondoskodik a programok megfelelő elhelyezéséről a felső memóriában, de ha ez nekünk nem felel meg, akkor a csomagban található analizáló program segítségével felderíthetők a szűk keresztmetszetek, tehát még utólag is korrigálhatunk. Avagy a Windows esetén elővehetjük a kézikönyveket (ha van!), és szülhetjük-izzadhatjuk ki magunkból a megfelelő beállításokat (ha sikerül!). Mellesleg a Manifest program részletesebb és pontosabb információkat ad, mint a Windows MSD.EXE programja.

Miért hagytam meg viszont a 386MAX program QCACHE programját? Furcsa, de igaz, hogy a SMARTDRV.EXE program nem gyorsítja az adatátvitelt. Ezt világosan kimutatják azok a programok, amelyek ezt a paramétert hivatottak mérni (például Checkit és Core). A SMARTDRV.EXE összegyűjti a DOS-hoz forduló programoktól a diszkhozzáférési igényeket, majd az első adandó alkalommal egyszerre végrehajtja őket. A közismert Norton Commander alatti

gyors törléseknek ez az oka, de abban a pillanatban lelassul a program, amint nagy mennyiségű adatot akarunk másolni két lemezegység között.

A QCACHE program az általam ismert egyik leggyorsabb cache-program. Lassabb valamivel a 3.3-as DOS-ban található SMARTDRV.SYS programnál, de gyorsabb, mint a PC-cache, és gyorsítja a floppyról olvasást is. Az eredmény kb. 4-5 Mbájt/s olvasási sebesség a merevlemezről egy normál 386-os rendszerben. Ahhoz, hogy Windows alatt is gyorsítson, a CONFIG.SYS-be be kell írni a következő sort:

```
device=Qcache.win
```

Ehhez meg kell lennie a 386MAX csomagból a QCACHE.WIN állománynak. Azok, akiknek Trident 8900-as sorozatú VGA kártyájuk van, és a nagyobb felbontás mellett is használni szeretnék a 256 színű módok valamelyikét, ne nagyon dolgozzanak a Qcache-sel, mert a gyári szoftverek Windows alatt összeakadnak vele.

Norton Desktop for Windows

Ha már a Windowsnál tartunk, érdemes megnézni: más szoftverekkel mi a helyzet?

Bizonyára sokan kedvelik a Norton Desktop for Windows csomagot és annak kezelői felületét. Minden tekintetben figyelmet érdemlő program. Számomra sokkal kényelmesebb használni, mint az eredetit. Az összevont program- és fájlmenedzser tényleg egy íróasztal felületének benyomását kelti — igazán stílusos környezet a munkához. Sajnos van egy hibája. Nagyon unalmas a hosszas installálási procedúra, a rengeteg floppy beetetése a számítógépbe. Ráadásul ott kell ülni előtte és várni, mikor kéri a következőt. És ha valamit elrontunk, lehet kezdeni elölről az egészet.

Mennyivel egyszerűbb lenne, ha bemásolhatnánk az egészet a merevlemezre, és valamilyen trükkel rávennénk, hogy installálja onnan magát! Nos, pontosan erre jó a SUBST parancs. Ha létrehozunk egy virtuális A vagy B meghajtót, például a C meghajtó egyik ideiglenes alkönyvtárában, máris nyert ügyünk van. De ekkor felfedeztem az NDW hibáját: nem hajlandó tudomásul venni ezt a virtuális meghajtót. Szerencsére van megoldás. Ha nem dobtuk el az eredeti fájlmenedzsert, akkor azt elindítva — csodák csodája! — megtaláljuk virtuális A vagy B meghajtónkat. Így lényegesen gyorsabb lesz az installálás, és ha a gép befejezte a munkát, nyugodtan törölhető a virtuális floppyról minden.

Installálási türelemjáték

Sajnos eredeti állapotában nem mindegyik szoftver alkalmas a virtuális meghajtóról való installálásra. Átalakítás nélkül mehet a Windows, a Works for Windows, az NDW, a Word for Windows, de nem futtatható így a CorelDraw, az MS-Publisher és valószínűleg még számos egyéb program. Azokat, amelyek az első csoportba tartoznak, elegendő XCOPY-val bemásolni a virtuális meghajtó gyökérkönyvtárába, el kell indítani a Windowst, majd a fájlmenedzser segítségével az adott program SETUP vagy INSTALL programját, ezután néhány kérdés megválaszolása következik, és az egész folyamat automatikusan lefut.

A CorelDraw-nak hozzunk létre 13 alkönyvtárat DISK1... DISK13 neveken, majd a lemezek tartalmát sorrendben ezekbe másoljuk be, így a SETUP program a DISK1 alkönyvtárban lesz.

Az MS-Publishernek hozzunk létre SETUP, PROGRAM és HELP neveken alkönyvtárakat, ezekbe másoljuk be utána a lemezek tartalmát, majd a fenti módon indíthatjuk a SETUP alkönyvtárból a hasonnevű programot a Windows alatt.

Az MS-Publisherben is akad hiba. Például az, hogy a File menü „Insert Picture" rutinja hibásan olvassa a 256 színű TIFF állományokat. A megoldás itt az lehet, hogy TIFF állományainkat felhasználás előtt valamilyen más formátumra konvertáljuk (például BMP-re).

Mindenkinek sok sikert kívánok a leírtak megfelelő alkalmazásához.

Voczelka Ferenc



Kitöltötte és visszaküldte már az Alaplap közvéleménykutató kérdőívét?

Ha nem kapott április számot, vagy más okból nincs már meg az űrlapja, kérjen a kiadótól! (IDG, 1536 Budapest I., Márvány u. 17.)

Aki május 31-ig visszaküldi, akár számítógépet is nyerhet !

Fit for today

A nagy IBM-felhasználók jól tudják (és számítanak is rá), hogy évente egy alkalommal nagyszabású show keretében ismerkedhetnek meg az IBM legújabb fejlesztési eredményeivel. Idén Nizzában rendezték meg Fit for today néven a Personal System Forumot, ahová közel ezer felhasználót hívott meg az IBM. A világ minden részéről idesereglett érdeklődőket nemcsak előadásokkal „bombázta" a kék óriás, hanem a kétnapos felhasználói fórumhoz kapcsolódó kiállításon meg is jelentek azok a termékek, amelyek a közeljövőben (a következő hetekben, hónapokban vagy év végén) jelennek meg az IBM kínálatában.

Bemutatkoztak azok az új fejlesztések, amelyekkel a lapzártánk után megrendezett és néhány napja befejeződött budapesti Ifabón már a magyar felhasználók is megismerkedhettek: a Raid 5 technológiájú PS/2 Server 295, a PS/1 multimédiarendszer, a PS/1 minitornyos változata, a ThinkPad és a Value Point gépcsalád.

A nyáron már az IBM is forgalmaz Pentium processzoros gépeket, amelyek a Raid 5 technológiával párosítva jelentősen megnövelik a számítógépek gyorsaságát. A PS/2 LAN Management egyik új funkciója révén megoldható a szoftverek gyors installálása egy távoli szerverről, a szintén újdonság remote audit segítségével pedig „leltározni" lehet a számítógép hardverelemei között, s így pontosan kiszámítható egy új rendszerre való átállás költsége. Ugyancsak az év közepén lesznek kaphatók az ergonomikus monitorok (Flat Panel), valamint a digitális vezérlésű, 17 "-os New Auto Scanning Monitor.

Ősszel kerül a nagyközönség elé a PS/2 E modell, egy helytakarékos, kis irodákban is jól használható számítógép, amelynek monitorját akár a falra is felszerelhetjük, képernyője nem villog, nincs semmiféle sugárzás. Az egér itt is része a klaviatúrának, de a numerikus billentyűzetet már „leválasztották", így azt balkezesek is kényelmesen használhatják. Az elegáns kivitelű számítógépben 2-2 hálózati és diszkbővítőhely található, így a PCMCIA technológia szerint a PS/2 E Token Ring hálózatba köthető, s két 105 MBájtos ütésálló diszkkel is tuningolható.

A ThinkPad itthon már kapható 700-as modelljei mellett a gépcsalád toll-alapú változatai arattak nagy sikert a fórum résztvevői körében. A 710T modellekben is tükröződik az IBM-nek a többi gyártóétól eltérő filozófiája: nem a kézzel írt szövegek felismerésére helyezték a hangsúlyt a fejlesztők, hanem az alkalmazók (eladók, szervizesek, brókerek) munkáját könnyítő „kipipálásokat és beikszeléseket" részesítették előnyben.

Hosszan lehetne sorolni még azokat az újításokat, amelyekkel mind-mind a felhasználók kegyeit keresi az IBM. Egy biztos: a francia Riviérián megrendezett fórumon képet alkothatott egy IBM-felhasználó arról, hogy eddigi beruházásaihoz és fejlesztési terveihez milyen mértékben számíthat a kék óriásra a közeljövőben.

Sziebig Andrea

Nyelvünkben él a szoftver

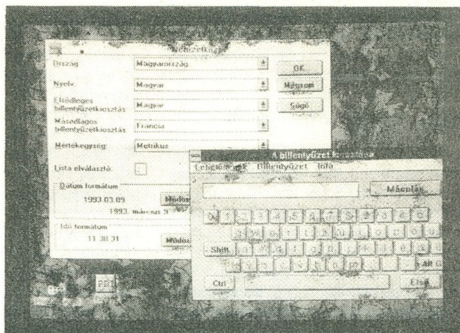
„Begyűrűdző" hardverek

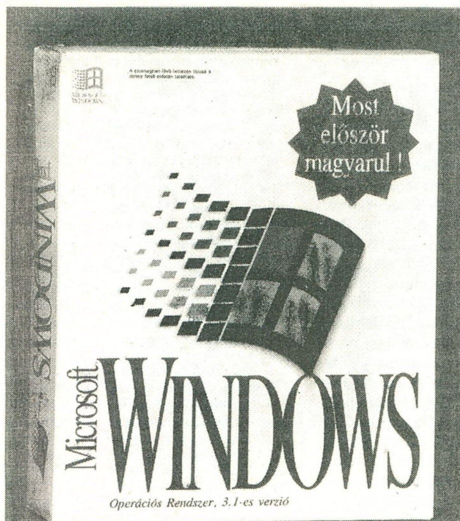
Hírcsokrunk nagyobbik felét ezúttal olyan szoftverek köréből válogattuk, amelyek mind tudnak magyarul (is). A magyar Windows színrelépése szolgáltatta az apropót ahhoz, hogy hírül adjuk: megszületett a cc:Mail levelezőrendszer és az Ami Pro szövegszerkesztő honi verziója.
Mielőtt azonban bárki „magyarkodással" vádolna minket, nemzetközi (hardver) vizekre evezünk: bemutatjuk a Silicon Graphics legújabb munkaállomásait, és bekukkantunk egy pénzkiadó automata gyomrába.

Jobb, mint az eredeti!

Régóta vártuk, hogy megjelenjen a már jóval korábban beharangozott magyar Windows. Márciusban azután visszavonhatatlanul megérkezett a Windows 3.1 magyar verziója. A Microsoft szerint ez azonban csak a kezdet, mert még ebben az évben számos más programjuk (Works for Windows, Word for Windows, Excel) bemutatkozik helyi változatban.

A magyar Windows többet tud az amerikainál(!), mert a magyar karakterek felhasználása mellett lehetővé teszi az összes európai nyelv karaktereinek alkalmazását. A Windows 3.1 magyar nyelvű változatának installálásakor meghatározhatjuk az elsődleges és a másodlagos billentyűkiosztást, a billentyűzetet megváltoztató folyamatot és a nyelvet. Alapértelmezés szerint az elsődleges a magyar, és a gyakran használt másik nyelvet célszerű a másodlagos kiosztáshoz beállítani. A magyar Windows része egy billentyűzetkijelző is, amely az elsődleges és a másodlagos közötti váltást teszi lehetővé. Ez a kijelző azt is megmutatja, hogy a kettő közül éppen melyik van használatban.





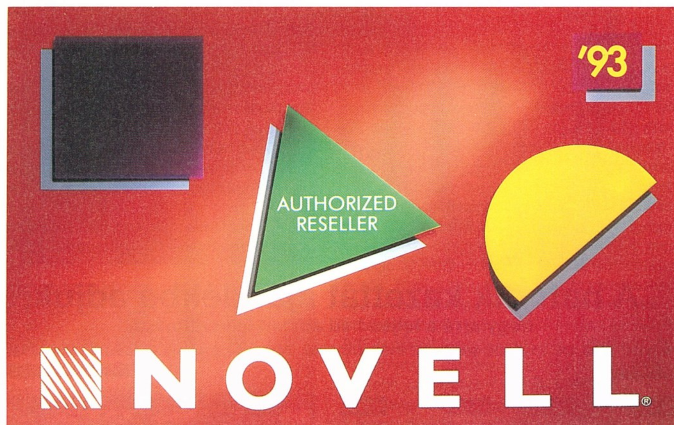
A magyar Windows alapértelmezés szerinti karakterkészletét az 1250-es kódlap foglalja magában. Ezt a karakterkészletet támogatják az új TrueType betűtípusok: Arial CE, Times New Roman CE, Courier New CE.

A Windows 3.1-es magyar verziójával, a TrueType betűtípusokkal többnyelvű dokumentumokat készíthetünk. Így például a Microsoft Word for Windows alkalmazásban létrehozhatunk egy olyan dokumentumot, amely magyar, orosz, angol és francia nyelvű részeket tartalmaz. Ahhoz, hogy ilyen többnyelvű közegben dolgozzunk, a Windows magyar változatát kell megvásárolnunk a Microsoft-partnereknél 12 000 forint (+ÁFA) körüli áron.

Levelek jönnek, levelek mennek...

A CeBIT-en örömmel láttuk, hogy a nagy szoftverházak kifejezetten törekszenek arra, hogy programjaik több nyelven beszéljenek. A Lotus különösen nagy gondot fordít erre a

Figyelje azonosító emblémánkat!



NetWare hálózati termékeket csak a Novell által feljogosított rendszerházaknál, dealereknél, viszonteladóknál vásároljon!

A fenti embléma alapján minden hálózat-felhasználó már az elsõ pillantásra felismerheti a Novell termékek forgalmazására valóban feljogosított rendszerházat, dealert, viszonteladót. Ezeknél a Novell által feljogosított cégeknél hálózat tervezést, szakszerű installálást és hatékony müszaki támogatást biztosítunk vásárlóinknak, felhasználóinknak. A NetWare termékek forgalmazására feljogosított cégeink az országban szinte mindenütt megtalálhatók. Érdeklődésére készséggel megadjuk az Önhöz legközelebb eső cég címét. Kérdésével forduljon a Novell disztributoraihoz:

INFORMÁCIÓKÉRÉS: A0501

Ez a feljogosított viszonteladók névsora Magyarországon 1993-ban.

Albacomp Rt.
Alfadat Kft.
Amicus Computer Bt.
Beks Bt.
Computer Praxis Kft.
Computer Technika Kft.
Controll Rt.
Daten-Kontor
Data Elektronik Kft.
Digital Kft.
Dimenzio Kisszövetkezet
DNC
Electrocoop Ltd.
ESCOM Computer
Euronet Kft.
Főnixcomp Computer Kft.
Humánsoft Kft.
IFSZ
Interface Kft.
Kerorg Kft.
KFKI Direkt Kft.
KFKI Computer Networks Ltd.
KFKI Xeus
Kontrax Irodatechnika
Lézer Elektronik Kft.
Makrotrend Kft.
MicroAge Békéscsaba
MicroAge Szeged
MicroAge Székesfehérvár
Microcomp sro.
Micronetwork Systems Kft.
MICROPO KSz.
Microsystem Rt.
MUTEX Kft.
Műszertechnika Computer Rt.
Műszertechnika Veszprém
Navigátor Kft.
Nádor Kft.
Netex Kft.
Orgware Kft.
Polygon Kft.
Rolitron Informatics
Rutinsoft Kft.
Summatech Kft.
SW Studio
Systrend Kft.
Szirt Computer Kft.
SZKI Rt.
Trade Hardszoft Kft.
Trigon Hardware Kft.
Unicomp Kft.
X-BYTE
Zalaszám Kft.
2R Periférla Kft.

3Soft Computer & Trading Ltd.
H-1123 Budapest
Tel. +36-1-156-5419

Computer 2000
H-1027 Budapest
Tel. +36-1-202-4520

Duna Elektronika Rt.
H-1083 Budapest
Tel. +36-1-267-1092

Selectrade Computer
H-1141 Budapest
Tel. +36-1-252-6130

Walton Networking Ltd.
H-1077 Budapest
Tel. +36-1-122-1846



The Past, Present and Future of Network Computing.

Amikor 5 kazetta hat

Vegyen 5 streamer-kazettát adatai lementésére hétfőtől-péntekig



A hatodik kazettát ajándékba adjuk,

hogy arra a teljes heti adatait lementhesse.

Akciónk a **DC 2000, DC 2120, DC 600 A, DC 6150, DC 6250, DC 6525** kazettákra vonatkozik, amíg a készlet tart.

További részletek forgalmazóinktól:

Albacomp Székesfehérvár ☎ (06-22) 315414
Digitech Szekszárd ☎ (06-74) 16874
Kventa Budapest ☎ 132-8112
Megoldás Kaposvár ☎ (06-82) 11646
Mixim Budapest ☎ 133-3500
RT-Trading Szeged ☎ (06-62) 325470
Summatech Győr ☎ (06-96) 18915
Tanker Budapest ☎ 183-0123
Texim Miskolc ☎ (06-46) 359294
B 30 Budapest ☎ 118-8881
Galax Budapest ☎ 161-0857
Macroda Budapest ☎ 156-4802
Microlan Debrecen ☎ (06-52) 43103
Műszertechnika Budapest ☎ 147-1590
S+H Mercurius Budapest ☎ 142-6172
Tabula Nagykanizsa ☎ (06-92) 73211
Telecomp Pécs ☎ (06-72) 11751
Volánelektronika Budapest ☎ 186-8818

Az innováció Önnek dolgozik™

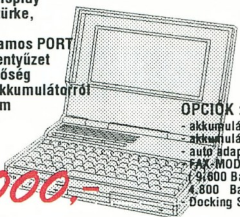
3M Hungária Kft.
1133 Budapest, Váci út 110.
Tel.: 267-1680, 267-1683 Fax: 267-1803

INFORMÁCIÓKÉRÉS: A0353 ▶

AUVA

888/25 NOTEBOOK

- 80386SXL-25 CPU
- 4 MB RAM (8 MB-ig bővíthető)
- 1,44 MB 3.5" FDD
- 80 MB 2.5" HDD
- 10"-os VGA LCD display (640 X 480, 64 szürke, levehető)
- 1 soros, 1 párhuzamos PORT
- külső VGA és billentyűzet csatlakozási lehetőség
- 2 órás működés akkumulátorról
- 280 X 225 X 45 mm
- 2,6 kg



OPCIÓK:
- akkumulátor töltő
- akkumulátor
- auto adapter
- FAX-MODEM (9.600 Baud adás, 4.800 Baud vétel)
- Docking Station

129.000,-

Árunk az ÁFA-t nem tartalmazza !

Multimedia

AUVA MEDIA

CD-ROM OLVASÓ
INSTALLÁLÓ LEMEZ
CD-TITLES
FELHASZNÁLÓI KÉZIKÖNYV
CSATLAKOZÓ KÁBEL
SZTEREÓ HANGKÁRTYA
HANGSZÓRÓK

AUVA MEDIA Upgrade Kit A66SP

SONY CD-ROM OLVASÓ
beépített nagy teljesítményű CD-ROM olvasó, adat- és hanghordozó CD olvasó

16-bit-es SZTEREÓ HANGKÁRTYA
Sampling Rate up to 44,1 KHz, hang szintetizátor, sztereó erősítő, MIDI/Joystick port, sztereó be- és kimenet, Sound Blaster pro Sound Card, Sound Blaster pro Software

SZTEREÓ HANGSZÓRÓK

SOFTWARE CSOMAGOK
2 Software csomag CD-ROM-on
- World Atlas for Windows
- Wing Commander / Ultima VI (Games)



makrotrend
ELEKTRONIKAI ÉS SZÁMÍTÁSTECHNIKAI SZÖVETKEZET

1143 Budapest, XIV. Hungária Krt. 65.-67.
Tel.: 183-43-56 Fax: 163-78-88

INFORMÁCIÓKÉRÉS: A0530 ▶

King Devran Travel

Velünk nemcsak Ön, a családja is meg lesz elégedve!

Tengerparti nyaralások

Spanyolország: Costa Brava 10 nap, 7 éjszaka, 3 csillagos hotel, félpanzió, luxusbusszal már 17900 forint

Olasz Riviéra: 8 nap szállodában, félpanzióval 16900 forint

Török Riviéra: 10 nap szállodában, félpanzióval 23900 forint

SZENZÁCIÓ!

Srí Lanka: 17 nap félpanzióval 109900 forint
Maldív-szigetek: 18 nap félpanzióval 119900 forint

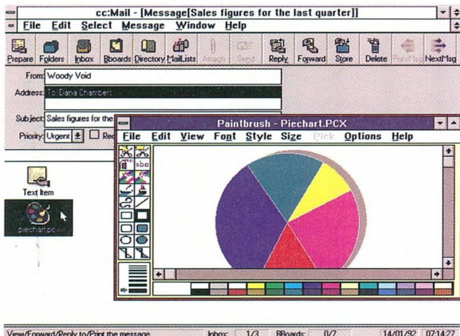
KING DEVRAN TRAVEL
Külkereskedelmi, Idegenforgalmi és Szolgáltató Kft.
1074 Budapest, Rákóczi út 64. • Telefon: (36-1)121-0913

INFORMÁCIÓKÉRÉS: A0516 ▶

területre. 8 nyelvre fordították le például a cc:Mail elektronikus levelezőrendszert, amelynek DOS alatt működő változata már tudott magyarul, a windowsos verzió honosítását pedig nemrég fejezték be a KFKI Számítógéphálózatok Kft szakemberei.

A magyar viszonyokra talán legjobban alkalmazható cc:Mail gyors, pontos, és szinte valamennyi levelezőrendszerrel (AT&T, ...) kapcsolatot tud teremteni. A lokális hálózatok piacán közel kétmillióan használják a világon, hatezren Magyarországon. (E hazai számadat nem sokkal marad el a közimert ELLA levelezőrendszer felhasználóinak száma mögött.)

A DOS, Windows, OS/2, Macintosh és Unix alapú cc:Mail levelezőrendszerek X.25-ön keresztül is összekapcsolhatók,



és faxüzeneteket is továbbíthatunk a cc:Mail központokon keresztül: az otthoni számítógépünkön megírt faxunkat elküldjük a központnak, ahonnan az akár fax-, akár más formátumban továbbítja levelünket.

A send ikonnal küldhetünk el levelet — egyvalakinek vagy körlevélként, de választhatunk címlistáról is. Beállíthatjuk levelünk tárgyát, sürgősségét, s kérhetünk visszaigazolást arról, hogy a „túloldalon" valóban elolvasták-e üzenetünket. Kényelmes, hogy a leveleket a már jól megszokott szövegszerkesztőnkkel is elkészíthetjük, de használhatjuk a cc:Mail saját szövegszerkesztőjét.

A levelezés azonban csak akkor igazán élvezetes, ha nemcsak küldünk, hanem kapunk is leveleket. Folyamatosan figyelhetjük, hogy jött-e üzenetünk. A programban egy zárt boríték jelzi a nem olvasott üzeneteket, nyitott az olvasottakat. Nem muszáj azonnal elovasnunk a levelet: két kattintással belenézhetünk, hogy kitől jött üzenet, s kik kapták még ugyanezt a levelet.

S ha kíváncsiságunkat felkeltette a feladó, akkor a levelet átolvasva akár rögtön válaszolhatunk rá, kiegészítve továbbküldhetjük másoknak, elmenthetjük archív fájlba vagy iratgyűjtőbe.

Az eredeti angol változattal teljesen kompatibilis magyar cc:Mail helpje is magyarul beszél. Az eltérő hálózatok, operációs rendszerek és számítógépek problémamentes kapcsolatát megvalósító, egyszerűen kezelhető program — a kis felhasználói csoportoktól a nagyvállalatokig — valamennyi szervezet kommunikációs igényét kielégíti.

S teszi mindezt a cc:Mail szélese körben is lérhető áron: a DOS-os verzió kb. 17 000 Ft, míg a windowsos változat kb. 42 000 Ft.

Egy vizuális szövegszerkesztő

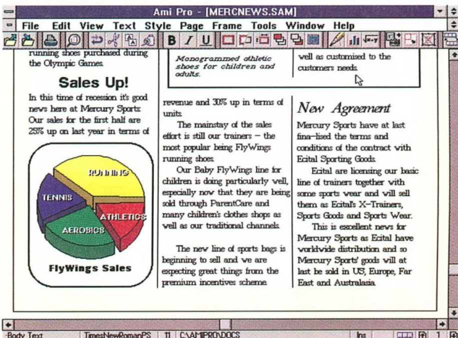
A Windows előretörésével egyre több Windows alatti szövegszerkesztő közül választhatunk. A rendkívül közkedvelt Word for Windows és a Wordperfect for Windows mellett nemrég végre megjelent egy magyarul is író szövegszerkesztő: az Ami Pro. Az SZKI a Lotus cég szoftverének honosításához a Windows 3.1 magyar változatát vette alapul.

Az Ami Pro egy vizuális szövegszerkesztő, vagyis amit a képernyőn látunk, egy az egyben kinyomtathatjuk. A beépített rajzoló és grafikonkészítő funkciókkal dokumentumaink színesebbé, adataink érthetőbbé válnak. A negyvennél több előre gyártott stíluslappal a célnak leginkább megfelelő dokumentumok (levelek, feljegyzések, faxok, újságcikkek,...) készíthetők. Ezek a stíluslapok természetesen nem örök érvényűek, igényeinknek megfelelően módosíthatjuk őket. A maximum 8 hasábos szövegek elkészítéséhez az Adobe Type Managerből és tíznél több fontból választhatunk, és az adatok, képek és szövegek exportja/importja simán megoldható a legelterjedtebb szövegszerkesztők körében.

A program használatát megkönnyíti a legördülő menük, a képernyő alján elhelyezkedő állapotsor, a súgó (Help) és az „okosikonok" (smarticon) használata. Az okos ikonokat bárhol elhelyezhetjük a képernyőn (nincs fix helyük), grafikus ábrával jelzik a hozzájuk rendelt funkciókat.

Az Ami Pro lehetőséget ad gyors formázásra: egy kiemelt szöveget az okos ikon szerinti szövegformátumra cserélhetünk. A program egy másik kényelmes szolgáltatása az előzetes információkérés, ahol előre megnézhetjük, hogy mi is van egy dokumentumban. Az Ami Próval külön keretben kezelhetjük a grafikákat, képeket, sőt a képekbe szöveget is írhatunk.

A dokumentum készítése közben a program helyesírásunkat is ellenőrzi. A beépített Helyes-e? program nemcsak az elütéseket vizsgálja, és javítja helyesírásunkat, hanem real time üzemben a szükséges elválasztásokat is végrehajtja.



Az Ami Pro eddigi tulajdonságaiból jól látszik, hogy a program egy hagyományos szövegszerkesztőnél kicsit többet, egy kiadványszerkesztőnél kicsit kevesebbet tud. A Windows alatti, 35 000 Ft körüli szoftvert bátran ajánljuk az angolul nem tudó laikus felhasználóknak — ők vannak többségben. A forgalmazó SZKI — ráadásként — egy sűrített magyar nyelvű kézikönyvvel segíti az amúgy is felhasználóbarát program elsajátítását.



Grafikában — verhetetlen

A CeBIT egyik jellegzetességének éreztük, hogy a CAD/CAM alkalmazások túlnyomó része vagy Silicon Graphics vagy Hewlett Packard gépeken futott. Különösen figyelemreméltó volt, hogy tavalyhoz képest mennyire megerősödött a Silicon Graphics, hiszen négy hatalmas bástya körül, tekintélyes méretű kiállítási területen vonultatta fel legújabb gépeit. A számítógépes grafika területén a grafikus munkaállomások és szerverek széles skáláját kínálta. Az év elején jelent meg a hazai piacon a Silicon Graphics legújabb termékcsaládja: a Challenge. A 64 bites, 100/150 MHz órajelű, R4400 processzorra épülő rendszer akár 36 processzorig bővíthető. Az elsősorban szerverfunkciókra kihegyezett, multiprocesszoros Challenge mellett egy lebegőpontos számításokra specializált termékkel is kirukkolt a Silicon Graphics. A Challenge-dzsel azonos architektúrájú Power Challenge 18 processzorig bővíthető, processzoronként 300 MFLOP számítási csúcsteljesítményre képes. A disztibútori feladatokat ellátó Creative Engineering Kft. szerint a termékcsalád rugalmasan konfigurálható, kedvező ár/teljesítmény mutatói alapján egy közepes pénzügyi kerettel rendelkező felhasználó számára sem elérhetetlen álom.

Az újonnan piacra dobott rendszerek nemcsak szerverként, hanem nagy teljesítményű grafikus munkahelyként is konfigurálhatók. Az új, multiprocesszoros grafikus rendszer (Onyx) kiemelkedő grafikai teljesítményét a VTX és a RealityEngine2 opciók biztosítják, amelyek a világ leggyorsabb(!) grafikus alrendszereinek számítanak.

A munkaállomások „jóságát” azonban igazából az alkalmazásokon keresztül lehet megítélni. A Silicon Graphics eredményeit különösen figyelemreméltónak tartjuk, mert például az Ars Electronica kiállítás valamennyi díjnyertes szoftvere az ő munkaállomásain futott. Azonban nemcsak az animáció területén sikeres a Silicon Graphics, hanem erős mérnöki tervezésben (CAD/CAM) és molekulamodellezésben.

Pénzfelvétel — falon át

Egy készpénzhez szokott országban — mint Magyarország — egyre nagyobb vehemenciával igyekeznek meggyőzni az embereket a pénzkímélő módszerek elterjesztői. Ezt láttuk a BankTech kiállításon is, ahol valamennyi bankautomatagyártó (Bull, Olivetti, NCR, IBM) felvonultatta legújabb fejlesztésű kültéri és beltéri ATM-jét.

A hazai bankokban és „vidékén” a legtöbbet Bull bankjegykiadó automatákkal találkozunk. A család legfrissebb tagját (Bull Questar 3410) elsősorban bankfiókon kívüli üzemeltetésre tervezték. A mágneskártyával és smart carddal (aktív memóriakártyával) működő berendezés az ügyfelek számára tökéletes diszkréciót (és pénzt) biztosít a hét minden napján, 24 órában.

Az ATM központi egysége egy DOS alatt működő Zenith 325 SX PC (1 Mbájt RAM, 40 Mbájt harddiszk, 1,44 Mbájt floppy, 5 kártyahely). Az ATM ügyfélinterfésze (14"-os színes VGA monitor, 2x4 oldalsó funkcióbillentyű, 15 gombos fémbillentyűzet, ISO kártyaolvasó, nyugtanyomtató) esztétikus formatervezésű. A képernyő és a klaviatúra ütésbiztos, így nincs szükség a korábbi típusoknál alkalmazott vandal doorra.

A nagy megbízhatóságú bankjegykiadó modul másodpercenként 15 bankjegy kiadását engedélyezi. A mechanikusan zárható és szabványos kazettákban 3400 új (2000 használt) bankjegy helyezhető el.

Az önkiszolgáló automaták logikai védelmet és fizikai biztonságot garantálnak. A titkosító modul helyi védelmet nyújt, ellenőrzi a kényes információk áramlását, és offline esetben kezeli a személyi azonosító kódot.

Ha az automatákhoz online hálózaton keresztül, egy számítógépes banki rendszerbe illesztve férhet hozzá az ügyfél, akkor az már a hazai banki kultúra fejlődését jelenti. S akkor talán a bankok vásárlókedve is jobban megjön a 2,5 — 4 millió forintos automatákhoz.

Sziebig Andrea




abc
SALES REVENUE
50%
28%
25%
Nashua C302